

Title of thesis

**Feedback Linearization Techniques for Collaborative
Nonholonomic Robots**

I, **SALMAN AHMED** hereby allow my thesis to be placed at the Information Resource Center (IRC) of Universiti Teknologi PETRONAS (UTP) with the following conditions:

1. The thesis becomes the property of UTP.
2. The IRC of UTP may make copies of the thesis for academic purposes only.
3. This thesis is classified as

☐

Confidential

☒

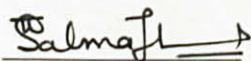
Non-confidential

If this thesis is confidential, please state the reason:

The contents of the thesis will remain confidential for _____ years.

Remarks on disclosure:

Endorsed by



Signature of Author



Signature of Supervisor

Permanent : House # 77, Street 4, Sector H4,
address Phase 2, Hayatabad,
Peshawar, NWFP, Pakistan.

Assoc. Prof. Dr. Mohd Noh Karsiti

Date : July 24th, 2007

Date : July 24th, 2007

UNIVERSITI TEKNOLOGI PETRONAS

Approval by Supervisor

The undersigned certify that he has read, and recommend to the Postgraduate Studies
Programme for acceptance, a thesis entitled

“Feedback Linearization Techniques for Collaborative Nonholonomic Robots”

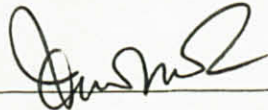
submitted by

Salman Ahmed

for the fulfillment of the requirements for the degree of

Masters of Science in Electrical and Electronic Engineering

Date: July 24th, 2007

Signature : 
Main Supervisor : Assoc. Prof. Dr. Mohd Noh Karsiti
Date : July 24th, 2007
Co-Supervisor 1 : _____
Co-Supervisor 2 : _____

UNIVERSITI TEKNOLOGI PETRONAS

Feedback Linearization Techniques for Collaborative Nonholonomic Robots

By

Salman Ahmed

A THESIS

SUBMITTED TO THE POSTGRADUATE STUDIES PROGRAMME

AS A REQUIREMENT FOR THE

DEGREE OF MASTERS OF SCIENCE IN ELECTRICAL AND ELECTRONIC

ENGINEERING

Electrical and Electronic Engineering

BANDAR SERI ISKANDAR,

PERAK

JUNE, 2007

DECLARATION

I hereby declare that the thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Teknologi PETRONAS or other institutions.

Signature : Salman Ahmed

Name : Salman Ahmed

Date : July 24th, 2007

ACKNOWLEDGEMENT

First and foremost, all praise and grace goes to Allah, without whose guidance, no effort can attain success.

I am indebted to my supervisor, Assoc. Prof. Dr. Mohd Noh Karsiti, for his guidance, inspiring discussions and support throughout this research. Without his never-ending patience, this thesis would probably not exist today. His useful suggestions and feedback have been the greatest help in the completion of this research.

I am thankful to Assoc. Prof. Dr. Herman Agustiawan and Mrs. Noor Hazrin Hany Hanif for their enlightening discussions, comments and help at several critical points in my research work. I would like to acknowledge Prof. Dr. Robert Loh of Oakland University, Prof. Dr. Jose B Cruz Jr. of Ohio State University and Mr. G M Hassan of Oklahoma State University for their encouraging and useful discussions.

I am grateful to the staff of the Research Enterprise Office. I am also deeply grateful to my fellow postgraduate and undergraduate students. They have been constant sources of encouragement and friendships during the course of this research.

I also wish to acknowledge the Government of Malaysia and Ministry of Science, Technology and Innovation for partially funding this project under the IRPA research grant scheme.

Finally, but most importantly, I want to thank my parents and family from the core of my heart. They have been a source of love, support, advice and feedback for me.

ABSTRACT

Collaborative robots performing tasks together have significant advantages over a single robot. Applications can be found in the fields of underwater robotics, air traffic control, intelligent highways, mines and ores detection and tele-surgery. Collaborative wheeled mobile robots can be modeled by a nonlinear system having nonholonomic constraints. Due to these constraints, the collaborative robots are not stabilizable at a point by continuous time-invariant feedback control laws. Therefore, linear control is ineffective, even locally, and innovative design techniques are needed. One possible design technique is feedback control and the principal interest of this thesis is to evaluate the best feedback control technique.

Feedback linearization is one of the possible feedback control techniques. Feedback linearization is a method of transforming a nonlinear system into a linear system using feedback transformation. It differs from conventional Taylor series linearization since it is achieved using exact coordinates transformation rather than by linear approximations of the system. Linearization of the collaborative robots system using Taylor series results in a linear system which is uncontrollable and is thus unsuitable. On the other hand, the feedback linearized control strategies result in a stable system. Feedback linearized control strategies can be designed based on state or input, while both state and input linearization can be achieved using static or dynamic feedback.

In this thesis, a kinematic model of the collaborative nonholonomic robots is derived, based on the leader-follower formation. The objective of the kinematic model is to facilitate the design of feedback control strategies that can stabilize the system and minimize the error between the desired and actual trajectory. The leader-follower formation is used in this research since the collaborative robots are assumed to have communication capabilities only.

The kinematic model for the leader-follower formation is simulated using MATLAB/Simulink. A comparative assessment of various feedback control strategies is evaluated. The leader robot model is tested using five feedback control strategies for different trajectories. These feedback control strategies are derived using cascaded system theory, stable tracking method based on linearization of corresponding error model, approximation linearization, nonlinear control design and full state linearization via dynamic feedback. For posture stabilization of the leader robot, time-varying and full state dynamic feedback linearized control strategies are used. For the follower robots using separation bearing and separation-separation formation, the feedback linearized control strategies are derived using input-output via static feedback.

Based on the simulation results for the leader robot, it is found that the full state dynamic feedback linearized control strategy improves system performance and minimizes the mean of error more rapidly than the other four feedback control strategies. In addition to stabilizing the system, the full state dynamic feedback linearized control strategy achieves posture stabilization. For the follower robots, the input-output via static feedback linearization control strategies minimize the error between the desired and actual formation. Furthermore, the input-output linearized control strategies allow dynamical change of the formation at run-time and minimize the disturbance of formation change. Thus, for a given feasible trajectory, the full state feedback linearized strategy for the leader robot and input-output feedback linearized strategies for the follower robots are found to be more efficient in stabilizing the system.

ABSTRAK

Robot-robot sekongkol mempunyai kelebihan dalam menjalankan kerja berbanding robot tunggal. Aplikasinya boleh ditemui dalam bidang robotik dasar laut, kawalan trafik udara, lebuh raya pintar, pengesanan galian dan bijih serta tele-bedah. Robot-robot sekongkol gerak beroda boleh dimodelkan dengan sistem tak linear dengan kekangan tak holonomi. Akibat kekangan-kekangan ini, robot-robot sekongkol tidak distabilkan pada satu titik dengan hukum kawalan suap balik tak berubah terhadap masa. Maka, kawalan linear, biarpun setempat, adalah tidak berkesan, membawa kepada keperluan terhadap kaedah reka bentuk teknik yang inovatif. Salah satu teknik reka bentuk adalah kawalan suap balik, dan tesis ini bertujuan untuk menilai teknik kawalan suap balik yang terbaik.

Pelinearan suap balik adalah salah satu teknik kawalan suap balik yang mungkin. Pelinearan suap balik adalah kaedah penjelmaan satu sistem tak linear kepada sistem linear menggunakan penjelmaan suap balik. Ia berbeza daripada siri pelinearan Taylor yang lazim, memandangkan ia dicapai menggunakan penjelmaan koordinat tepat berbanding anggaran linear bagi sistem. Pelinearan bagi sistem robot-robot sekongkol menggunakan siri Taylor menghasilkan sistem linear yang tak terkawal. Walau bagaimanapun, strategi suap balik linear menghasilkan sistem yang stabil. Strategi suap balik linear boleh direkabentuk berdasarkan keadaan atau input, dengan kedua-dua keadaan dan pelinearan input dapat dicapai menggunakan suap balik statik atau dinamik.

Model kinematik robot-robot sekongkol tak holonomi telah diterbitkan dalam tesis ini berdasarkan formasi ketua-pengikut. Objektif model kinematik ini adalah untuk memudahkan reka bentuk strategi kawalan suap balik yang dapat menstabilkan sistem dan meminimumkan ralat di antara trajektori yang dikehendaki dan trajektori sebenar. Formasi ketua-pengikut digunakan dalam kajian ini memandangkan robot-robot sekongkol diandaikan hanya mempunyai kebolehan komunikasi semata-mata.

Model kinematik untuk formasi ketua-pengikut disimulasikan menggunakan MATLAB/Simulink. Satu penilaian secara membandingkan pelbagai strategi kawalan suap balik telah dinilai. Model ketua robot diuji menggunakan lima strategi kawalan suap balik untuk trajektori yang berbeza-beza. Kesemua strategi kawalan ini diterbitkan menggunakan sistem teori terlata, kaedah pengesanan stabil berdasarkan pelinearan yang sepadan dengan ralat model, anggaran pelinearan, reka bentuk kawalan tak linear dan reka bentuk suap balik keadaan dinamik penuh terlinear. Pengubahan masa dan strategi kawalan suap balik keadaan dinamik penuh terlinear digunakan bagi penstabilan postur bagi robot ketua. Bagi robot-robot pengikut yang menggunakan pemisahan bearing dan formasi pemisahan-pemisahan, strategi kawalan pelinearan suap balik diterbitkan menggunakan input-output melalui suap balik statik.

Berdasarkan keputusan simulasi untuk robot ketua, didapati bahawa kawalan suap balik keadaan dinamik penuh terlinear memperbaiki prestasi sistem dan meminimalkan ralat secara mendadak berbanding keempat-empat strategi kawalan suap balik yang lain. Di samping menstabilkan sistem, strategi kawalan suap balik keadaan dinamik penuh terlinear mencapai penstabilan postur. Bagi robot-robot pengikut, strategi kawalan input-output melalui strategi pelinearan suap balik telah meminimumkan ralat di antara formasi kehendak dan sebenar. Tambahan pula, strategi kawalan pelinearan input-output membenarkan perubahan dinamik bagi formasi pada masa perjalanan dan meminimalkan gangguan perubahan formasi. Oleh itu, untuk satu-satu trajektori tersaur, strategi suap balik keadaan penuh terlinear untuk robot ketua dan strategi suap balik input-output terlinear untuk robot-robot pengikut didapati lebih berkesan dalam menseimbangkan sistem.

TABLE OF CONTENTS

STATUS OF THESIS	i
APPROVAL PAGE	ii
TITLE PAGE	iii
DECLARATION	iv
ACKNOWLEDGEMENT	v
ABSTRACT.....	vi
ABSTRAK.....	viii
TABLE OF CONTENTS.....	x
LIST OF TABLES	xiv
LIST OF FIGURES	xv
ABBREVIATIONS	xvii

CHAPTER 1: INTRODUCTION.....1

1.1 Collaborative Robots	1
1.2 Motion Planning.....	1
1.3 Problem Statement	3
1.4 Modeling and Control Analysis	4
1.5 Literature Review.....	6
1.6 Objectives & Scope.....	7
1.7 Thesis Organization	7

CHAPTER 2: MODELING OF COLLABORATIVE ROBOTS9

2.1 Holonomic and Nonholonomic Systems	9
2.2 Lie Bracket.....	11
2.2.1 Lie Bracket and Lie Bracket Tree	11
2.2.2 Involutive Distribution and Frobenius Theorem.....	13
2.3 Types of Modeling.....	13
2.4 Kinematic Modeling of Collaborative Nonholonomic Robots.....	15
2.5 Leader-Follower Formation for Collaborative Nonholonomic Robots.....	18

2.5.1	Separation Bearing Controller	19
2.5.2	Separation-separation Controller	21
2.6	Information Sharing among Collaborative Robots	22
2.7	Summary	26
CHAPTER 3: ANALYSIS OF CONTROL PROPERTIES		27
3.1	Controllability and Stability at a Point.....	27
3.2	Controllability and Stability around a Trajectory	30
3.3	Observability	32
3.4	Chained Form.....	35
3.5	Differential Flatness.....	37
3.6	Summary	37
CHAPTER 4: FEEDBACK LINEARIZATION TECHNIQUES.....		38
4.1	Mathematical Preliminaries	38
4.1.1	Equilibrium Point.....	38
4.1.2	Types of Continuous Function.....	39
4.1.3	Stability of Equilibrium Point.....	39
4.1.4	Lyapunov Stability.....	41
4.2	Feedback Control Design Techniques	42
4.2.1	Taylor Series Expansion	43
4.2.2	Feedback Linearization.....	44
4.3	Feedback Controller using Approximate Linearization.....	45
4.4	Feedback Controller using Cascaded Systems Theory	49
4.5	Feedback Controller using Linearization of Error Model.....	50
4.6	Feedback Controller using Nonlinear Design.....	51
4.7	Full State Feedback Linearization via Static Feedback	52
4.8	Input-output Feedback Linearization via Static Feedback.....	53
4.9	Full State Feedback Linearization via Dynamic Feedback.....	55
4.10	Posture Stabilization	59
4.10.1	Smooth Time-varying Feedback.....	59

4.10.2	Design based on Polar Coordinates	60
4.10.3	Dynamic Feedback Linearized Controller	62
4.11	Feedback Linearized Control Strategies for Follower Robots.....	63
4.11.1	Feedback Strategy for Separation Bearing Controller	63
4.11.2	Feedback Strategy for Separation-separation Controller	65
4.12	Summary	66
CHAPTER 5: SIMULATION RESULTS		67
5.1	Feedforward command controller.....	67
5.2	Framework for Collaborative Robots	68
5.3	Simulation Results for the Leader Robot.....	70
5.3.1	Test 1 (Eight Shaped Trajectory).....	71
5.3.2	Test 2 (Straight Line Trajectory)	76
5.3.3	Test 3 (Sinusoid Trajectory)	80
5.3.4	Test 4 (Circular Shaped Trajectory)	82
5.4	Discussion of Results for the Leader Robot	84
5.5	Simulation Results for the Follower Robots	93
5.5.1	Separation Bearing Controller	93
5.5.2	Separation-separation Controller	96
5.6	Discussion of Results for the Follower Robot	97
5.7	Posture Stabilization Controller	98
5.7.1	Time-varying Feedback Controller.....	98
5.7.2	Polar Coordinates Feedback Controller	100
5.7.3	Full State Linearized via Dynamic Feedback Controller.....	101
5.8	Discussion of Results for Posture Stabilization	103
5.9	Summary	103
CHAPTER 6: CONCLUSION.....		104
6.1	Accuracy and Stability of Feedback Controllers	104
6.2	Thesis Contribution.....	105
6.3	Future Work	106

PUBLICATIONS107

REFERENCES.....108

APPENDICES.....112

 Appendix A: M-file for calculating the Lie Bracket.....112

 Appendix B: M-file for checking the Linear Controllability.....113

 Appendix C: M-file for calculating the Jacobian Coefficients114

 Appendix D: Coordinate Transformation.....115

 Appendix E: M-file for calculating the Similarity Transformation116

LIST OF TABLES

Table 2.1:	Comparison of various wireless technologies.....	23
Table 2.2:	Comparison between various protocol suites	26
Table 5.1:	Parameters values for different feedback controllers.....	70
Table 5.2:	Error statistics using different feedback controller for test 1.....	74
Table 5.3:	Error statistics using different feedback controller for test 2.....	79
Table 5.4:	Error statistics using different feedback controller for test 3.....	81
Table 5.5:	Error statistics using different feedback controller for test 4.....	82
Table 5.6:	Effect of changing gains and parameters in feedback controllers	85

LIST OF FIGURES

Figure 1.1:	Point to point motion for two unicycle collaborative robots	2
Figure 1.2:	Trajectory tracking for two unicycle collaborative robots.....	3
Figure 2.1:	Nonholonomic wheeled mobile robot.....	10
Figure 2.2:	Lie Bracket tree.....	12
Figure 2.3:	Local and global frames for unicycle robot positioning	14
Figure 2.4:	Leader follower formation using separation bearing controller	19
Figure 2.5:	Leader follower formation using separation-separation controller.....	22
Figure 2.6:	PANU – PANU mode for communication among two robots	24
Figure 2.7:	GN mode for communication among robots	24
Figure 2.8:	NAP mode for communication among robots	25
Figure 4.1:	Leader robot positioning	45
Figure 4.2:	Leader robot kinematic model	47
Figure 4.3:	Feedback controller using approximate linearization.....	48
Figure 4.4:	Leader robot using approximate linearization feedback controller	49
Figure 4.5:	Feedback controller using cascaded systems theory.....	50
Figure 4.6:	Feedback controller using linearized of error model.....	51
Figure 4.7:	Feedback controller using nonlinear design	53
Figure 4.8:	Full state linearized via dynamic feedback controller	59
Figure 4.9:	Leader robot positioning in polar coordinates	60
Figure 4.10:	Posture stabilization controller based on polar coordinates.....	61
Figure 4.11:	Posture stabilization controller based on dynamic feedback	62
Figure 4.12:	Input-output linearized feedback controller for follower robot using separation bearing formation	65
Figure 4.13:	Input-output linearized feedback controller for follower robot using separation-separation formation.....	66
Figure 5.1:	Feedforward command controller for the leader robot.....	68
Figure 5.2:	Framework for the leader robot	68
Figure 5.3:	Framework for the follower robot.....	69
Figure 5.4:	Non-feasible trajectories for nonholonomic robots	69

Figure 5.5: Desired trajectory for test 1.....71

Figure 5.6: Linear velocity for the desired trajectory of test 171

Figure 5.7: Angular velocity for the desired trajectory of test 172

Figure 5.8: Norm of error for the trajectory of test 1 using approximate linearized controller73

Figure 5.9: Norm of error for the trajectory of test 1 using nonlinear controller73

Figure 5.10: Norm of error for the trajectory of test 1 using cascaded systems controller73

Figure 5.11: Norm of error for the trajectory of test 1 using stable tracking controller74

Figure 5.12: Norm of error for the trajectory of test 1 using full state linearized via dynamic feedback controller.....74

Figure 5.13: Actual trajectory by the leader robot for test 1 using full state via dynamic feedback linearized and nonlinear controller75

Figure 5.14: Desired trajectory for test 2.....76

Figure 5.15: Norm of error for the trajectory of test 2 using approximate linearized controller77

Figure 5.16: Norm of error for the trajectory of test 2 using nonlinear controller77

Figure 5.17: Norm of error for the trajectory of test 2 using cascaded systems controller78

Figure 5.18: Norm of error for the trajectory of test 2 using stable tracking controller78

Figure 5.19: Norm of error for the trajectory of test 2 using full state linearized via dynamic feedback controller.....79

Figure 5.20: Desired trajectory for test 3.....80

Figure 5.21: Desired and actual trajectory using full state linearized via dynamic feedback controller for test 381

Figure 5.22: Desired trajectory and actual trajectory using cascaded systems controller for test 4.....83

Figure 5.23: Actual trajectory using approximate linearized and Lyapunov function controller for test 483

Figure 5.24:	Actual trajectory using nonlinear and full state linearized via dynamic feedback controller for test 4	84
Figure 5.25:	Effect of ζ on mean of error using approximate linearized controller ..	86
Figure 5.26:	Effect of b on mean of error using approximate linearized controller	86
Figure 5.27:	Effect of ζ on mean of error using nonlinear controller	87
Figure 5.28:	Effect of b on mean of error using nonlinear controller	87
Figure 5.29:	Effect of c_1 on mean of error using cascaded systems controller	88
Figure 5.30:	Effect of c_2 on mean of error using cascaded systems controller	88
Figure 5.31:	Effect of c_3 on mean of error using cascaded systems controller	89
Figure 5.32:	Effect of K_x on mean of error using stable tracking controller	89
Figure 5.33:	Effect of K_y on mean of error using stable tracking controller	90
Figure 5.34:	Effect of K_θ on mean of error using stable tracking controller	90
Figure 5.35:	Effect of k_{p1} on mean of error using full state linearized via dynamic feedback controller	91
Figure 5.36:	Effect of k_{p2} on mean of error using full state linearized via dynamic feedback controller	91
Figure 5.37:	Effect of k_{d1} on mean of error using full state linearized via dynamic feedback controller	92
Figure 5.38:	Effect of k_{d2} on mean of error using full state linearized via dynamic feedback controller	92
Figure 5.39:	Actual trajectory using nonlinear and full state linearized via dynamic feedback controller for test 4	93
Figure 5.40:	Actual trajectory using separation bearing controller for test 1	94
Figure 5.41:	Actual trajectory using separation bearing controller for test 4	94
Figure 5.42:	Separation distance for the follower robot using separation bearing controller	95
Figure 5.43:	Separation angle for the follower robot using separation bearing controller	95
Figure 5.44:	Actual trajectory using separation bearing controller for test 4	96
Figure 5.45:	Actual trajectory using separation-separation controller for test 1	97

ABBREVIATIONS AND NOMENCLATURE

DOF: Degrees of Freedom

PANU: Personal Area Network User

GN: Group Adhoc-Network

NAP: Network Access Point

MIMO: Multiple Input Multiple Output

FIPA: Foundation for Intelligent Physical Agents

ACL: Agent Control Language

\forall : for all

$\|p\|$: norm of vector p

CHAPTER 1

INTRODUCTION

1.1 Collaborative Robots

Collaborative robots involve a team of robots to achieve a task. Collaborative robots cooperating with each other to solve problems can provide more capabilities than a single robot. Applications of collaborative robots can be found in the fields of underwater robotics, air traffic control, intelligent highways, security patrols, tele-surgery and mines and ores detection [1], [2], [3], [4]. The accuracy of task completion by collaborative robots depends on the control strategies implemented in the robots to minimize the effect of external disturbances and errors.

Collaborative unicycle robots can be represented as an underactuated system in which the number of control inputs is less than the number of generalized coordinates. In such a system, the controllable degrees of freedom (DOF) is less than the total degrees of freedom. Hence, motion control problems for such systems have attained considerable attraction over the past few years [5].

1.2 Motion Planning

Motion planning for nonholonomic collaborative robots involves the following basic motion tasks.

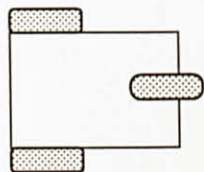
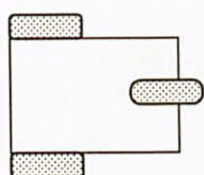
- **Point to Point Motion**

In point to point motion, the collaborative robots must reach a final goal starting from a given initial configuration. The trajectory or path for the collaborative robots is not specified in advance.

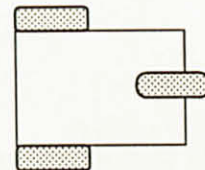
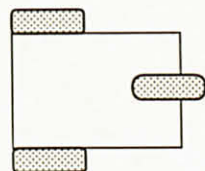
- **Trajectory Tracking**

In trajectory tracking, the collaborative robots must reach a final configuration following a certain desired trajectory in the cartesian space. The desired trajectory is a function of time. The collaborative robots' starting position can be either a part or not a part of the desired trajectory.

For an obstacle-free environment, point to point motion and trajectory tracking are shown in Figures 1.1 and 1.2, respectively.



Starting point



Goal point

Figure 1.1: Point to point motion for two unicycle collaborative robots

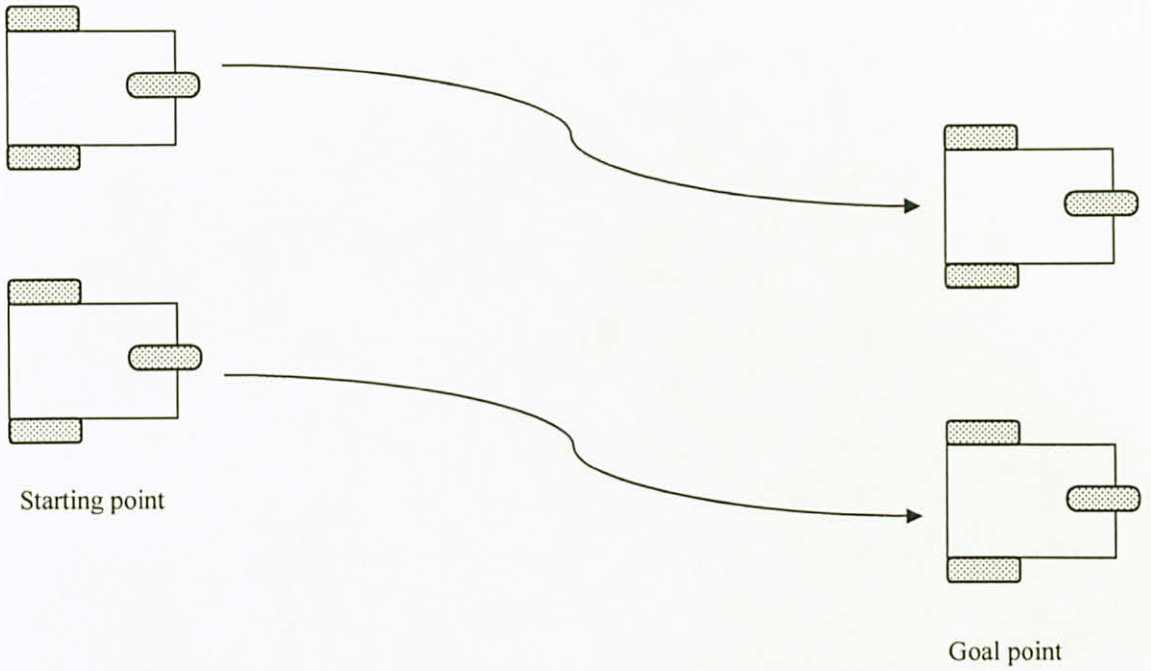


Figure 1.2: Trajectory tracking for two unicycle collaborative robots

In terms of control systems, point to point motion can be compared with a regulation control or posture stabilization problem for an equilibrium point in the state space. Trajectory tracking can be compared with a tracking problem such as to minimize the error between the reference and desired trajectory to zero.

1.3 Problem Statement

Collaborative wheeled mobile robots can be modeled as a nonlinear system with nonholonomic constraints imposed on their kinematics. Due to the nonholonomic constraints, the collaborative unicycle robots are not stabilizable at a point by smooth continuous time-invariant feedback control laws, according to Brockett theorem [6]. The theorem states that for a system to be smooth stabilized, the number of inputs must be equal to the number of states.

Let m and n represent the dimension of input and output, respectively. In case of point to point motion, m is two and n is three. The point to point motion for the robots implies zeroing three independent variables. In case of trajectory tracking, m is two and n is two. The objective of trajectory tracking is to stabilize the two dimensional error vector associated with the cartesian trajectory to zero. Thus, point to point stabilization and controllability cannot be achieved using linear control, even locally, and innovative design techniques are required. One such possible design techniques is feedback linearization and this is the principal investigation of this thesis.

Feedback linearization is the method of transforming a nonlinear system into a linear system (fully or partially) via a coordinate transformation, referred to as feedback transformation [7], [8]. Feedback linearization differs from conventional linearization, such as Jacobian linearization, because feedback linearization is achieved using exact state transformations and feedback, rather than by linear approximations.

1.4 Modeling and Control Analysis

Collaborative robots need to maintain a certain formation control to complete a task. The various approaches to formation control can be divided into three categories: behavior-based, virtual structure and leader-follower formation [9]. The behavior-based formation is a distributed approach and relies on implicit communication between robots [10]. The virtual structure is a centralized approach [11]. Majority of current algorithms that focus on behavior-based and virtual structure formation are implemented on robots having visual capabilities [12], [13].

In this thesis, the robots are assumed to have communication capabilities only. Therefore, the above two formation control approaches are not suitable; instead, the leader-follower formation is used in this research. The leader-follower formation is a centralized approach in which all the robots have a common goal. One of the robots is designated as the leader and the others become the follower robots. The leader robot follows a desired trajectory and guides the formation, while the follower robots follow the leader robot.

Using the leader-follower formation, the kinematic model for the collaborative robots system is derived. After deriving the kinematic model, the control properties including controllability, stability and observability are analyzed for the system. The collaborative robots are not linearly controllable and stable around a point. Therefore, tools from nonlinear control theory are used [14]. Lie Bracket is one such tool with the help of which, the control properties of collaborative robots can be analyzed. Using the Lie Bracket expansion, the collaborative robots are nonlinearly controllable and stable around a point.

After analyzing the control properties, the system is transformed into chained forms. Chained forms are canonical model structures which allow simple implementation of the system consisting of integrators. The chained form transformation is useful in finding the flat outputs of the system. The flat outputs are used to generate the desired reference trajectory. The reference trajectory is given as an input to the feedforward controller to generate the control inputs for the leader robot. These control inputs are transmitted to the follower robots using the Bluetooth piconet profile. The follower robots derive their own control inputs based on the inputs sent by the leader robot.

The control inputs for the leader and the follower robots are fed into the feedback linearized controller, which generates the actual inputs for the robots. The actual inputs are calculated based on the current states of the robot and the inputs from the feedforward controller. The objective of the feedback controller is to minimize the error between the actual and the desired trajectory. Furthermore, the feedback controller must achieve posture stabilization.

The purpose of this thesis is to study the issues related to nonlinear controllability, stability and the design of feedback controllers for collaborative nonholonomic robots.

1.5 Literature Review

The feedback linearized control design for collaborative robots is two-folds; first to design feedback law for the leader robot and secondly to design feedback laws for the follower robots. The leader robot is a single robot and is modeled using the unicycle robot system. The follower robots are also unicycle robots but using the leader-follower formation, they are modeled relatively to the leader and formation.

There have been various approaches to designing feedback control laws for the leader robot. A nonlinear feedback controller for formation control was proposed in [15]. The controller achieves asymptotic stability but the control laws depend on vision based inputs. A feedback control strategy was designed using cascaded system theory in [16]. The controller results in K - exponentially stable system and locally uniform exponentially stable system. A stable tracking controller based on the linearization of corresponding error model was proposed in [17]. The control laws result in locally asymptotic and locally uniformly asymptotic stable systems. A linear controller based on approximate linearization was proposed in [18]. The approximate linearized control laws result in a time-varying controller which does not guarantee asymptotic stability. A nonlinear controller for trajectory tracking was proposed in [19] which globally asymptotically stabilize the system. A dynamic feedback linearized control strategy was proposed in [20]. The dynamic feedback linearized control strategy results asymptotic tracking of the desired trajectory.

For the follower robots, feedback control strategies based on input-output linearized for the follower robots using separation bearing and separation-separation formations were presented in [21], [22], [23]. These feedback strategies stabilize the system and achieve the desired formation. For posture stabilization, a time-varying feedback controller was presented in [19]. A feedback controller based on polar coordinate transformation was presented in [24]. A dynamic feedback linearized controller was presented in [25] to achieve posture stabilization.

1.6 Objectives & Scope

The objectives and scope of this thesis are outlined below:

- To mathematically analyze the nonlinear control properties for collaborative nonholonomic robots.
- To allow the collaborative nonholonomic robots to share information using the Bluetooth piconet profile.
- To model the kinematics of nonholonomic collaborative robots using MATLAB/Simulink and simulate feedback control strategies.
- To develop a framework for collaborative nonholonomic robots using the leader-follower formation.
- To design feedback control strategies for the leader-follower formation using feedback linearization techniques.
- To analyze the performances of feedback control laws for collaborative nonholonomic robots.

1.7 Thesis Organization

This thesis is structured as follows. Chapter 2 presents the kinematic modeling of the leader-follower formation for collaborative robots. The concept of nonholonomy is discussed. To allow communication between robots, the Bluetooth piconet profile is discussed.

Chapter 3 presents the analysis of control properties for collaborative robots. The control analysis involves checking the controllability, stability and observability of the system. The system is also transformed into chained forms and the flat outputs for the systems are identified.

Chapter 4 presents the feedback controllers for collaborative robots using the leader-follower formation. The stability analysis of the feedback controllers is also presented.

The full-state and input-output linearized design techniques are applied to the leader-follower formation. The posture stabilization controllers for the leader robot are also presented.

Chapter 5 presents the simulation results for different feedback controllers discussed. The feedforward command controller is derived. A framework for collaborative robots is presented. The feedback linearized control strategies and posture stabilization controllers are simulated for a given set of trajectories. The error statistics using different feedback controllers are also presented.

Chapter 6 presents the conclusions. The contribution of the thesis and future work is also presented.

CHAPTER 2

MODELING OF COLLABORATIVE ROBOTS

This chapter presents the kinematic model for collaborative robots. The kinematic model for collaborative robots involves the concept of nonholonomy. In order to check whether a system is holonomic or nonholonomic, tools from nonlinear control theory will be used. Lie Bracket is one of the available tools that will be used for checking the controllability of the system. Finally, to allow communication among collaborative robots, the Bluetooth Personal Area Network (PAN) profile will be discussed.

2.1 Holonomic and Nonholonomic Systems

A kinematic constraint usually limits the position and/or the velocities in a system. A kinematic constraint can be holonomic or nonholonomic. A holonomic constraint is written as an equation independent of the generalized velocity vector as

$$f(p, t) = 0 \tag{2.1}$$

where $p \in X \subset \mathbb{R}^n$ represents the generalized coordinate vector or the state vector and n is the dimension of the state space, X . A holonomic constraint depends only on the coordinates of the system, p , and the time, t . It does not directly depend on the velocity or momentum of the system. A holonomic system may contain holonomic or no constraints. A mobile robot capable of moving in arbitrary directions is a holonomic system. A mobile robot capable of only translations is also holonomic. A system is nonholonomic if it cannot be written in the form of Eq. 2.1. Let \dot{p} represent the velocity vector, so a nonholonomic system can be expressed as

$$f(p, \dot{p}) = 0 \tag{2.2}$$

A nonholonomic system can not move in arbitrary directions in its configuration space. Examples of nonholonomic systems include cars, bicycles, unicycles, wheeled mobile robots, space robots, etc. A wheeled mobile robot can move in some directions (forward and backwards), but not others (sideward) directly as shown in Figure 2.1. To move sideways from position A to position B, the robot has to undergo a series of maneuvers via position C.

In general, for a system with n coordinates and k nonholonomic constraints, the allowable velocities are restricted to $m = n - k$ dimensional space. If a system has a kinematic constraint in which the velocities appear in the constraint equations, then the system is nonholonomic. There are various sources of nonholonomic constraints such as bodies in rolling contact without slipping (wheeled mobile robots or automobiles), angular momentum conservation and underactuated mechanical systems (having less control inputs than the number of states).

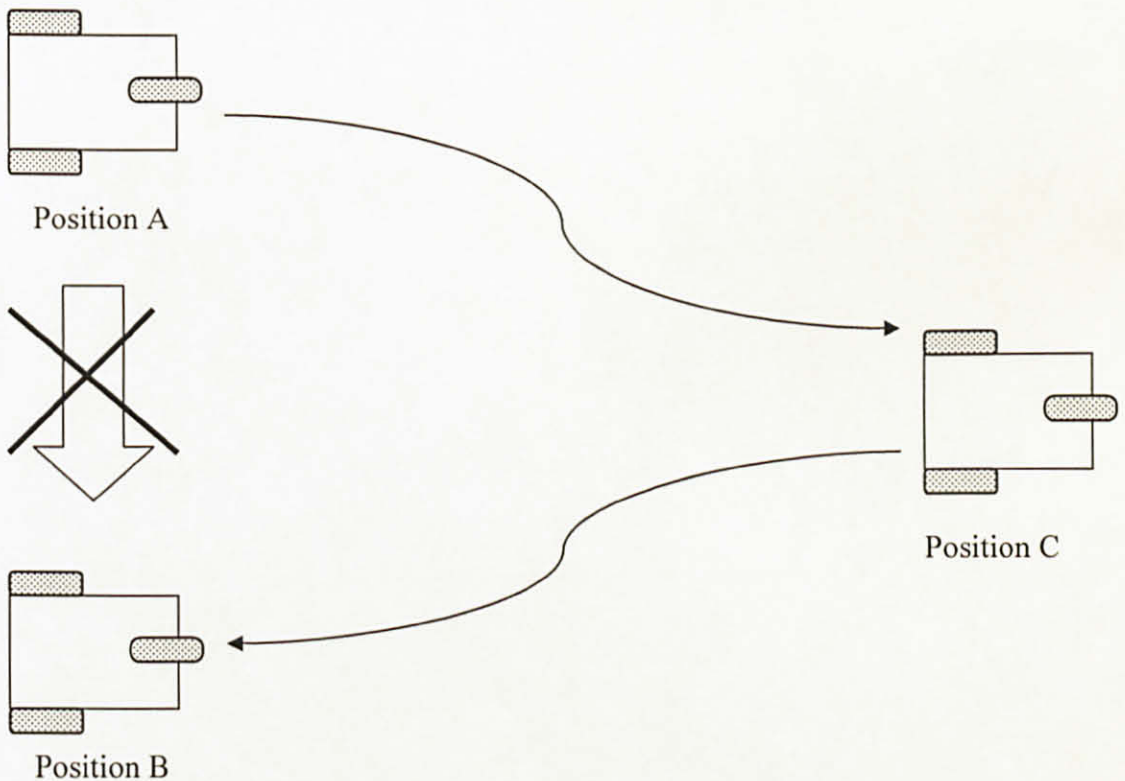


Figure 2.1: A nonholonomic wheeled mobile robot

All the possible configurations of a system can be determined by assigning a minimum number of independent parameters p_1, p_2, \dots, p_n which are called Lagrangian coordinates, where n represents the system degrees of freedom (DOF). A holonomic constraint is expressed as a function of Lagrangian coordinates and therefore it reduces the system DOF equal to the corresponding number of constraint equations. On the other hand, nonholonomic constraints, applying restrictions only to the velocities, do not prevent the attainment of any configuration and therefore do not lessen the system's DOF.

2.2 Lie Bracket

The Lie Bracket is the only tool required to determine whether a system is holonomic or nonholonomic [14]. A holonomic system is integrable but a nonholonomic system is not integrable. To check whether a system is integrable or not, its distribution, Δ , is computed using the Lie Bracket. If Δ is involutive, then the system is nonholonomic, otherwise it is a holonomic system.

2.2.1 Lie Bracket and Lie Bracket Tree

Let $x \in \mathbb{R}^n$ represent the state of the system, $u_i \in \mathbb{R}^m$ represent the control inputs and n and m represent the dimension of the state space and control inputs, respectively. A system, $\dot{x} = f(x, u)$ can be written in the form of

$$\dot{x} = g_o(x) + \sum_{i=1}^m g_i(x) u_i \quad (2.3)$$

where g_o represents the drift terms and each $g_i \in \mathbb{R}^n$ represents the system vector fields. For a driftless system, Eq. 2.3 can be expressed as

$$\dot{x} = \sum_{i=1}^m g_i(x) u_i \quad (2.4)$$

Let $g_1(x)$ and $g_2(x)$ denote two system vector fields. The associated Lie Bracket of $g_1(x)$ and $g_2(x)$ is denoted by $[g_1, g_2]$ and is defined as

$$[g_1, g_2] = \frac{\partial g_2}{\partial x} g_1 - \frac{\partial g_1}{\partial x} g_2 \quad (2.5)$$

The Lie Bracket is anti-commutative. Using the properties of Lie algebra [14], the Lie Bracket satisfies the property of skew symmetry which is

$$[g_1, g_2] = -[g_2, g_1] \quad (2.6)$$

The Lie Bracket tree is a tree formed by successive nested computations of the Lie Brackets as shown in Figure 2.2. Using the property of skew symmetry expressed by Eq. 2.6, the left branch of the Lie Bracket tree is equal in magnitude to the right branch but there is a difference of sign change. If a system has m system vector fields and can be expressed as of the form of Eq. 2.4, then there are $\binom{m}{2}$ Lie Brackets of the form $[g_i, g_j]$ for $i < j$ that can be formed.

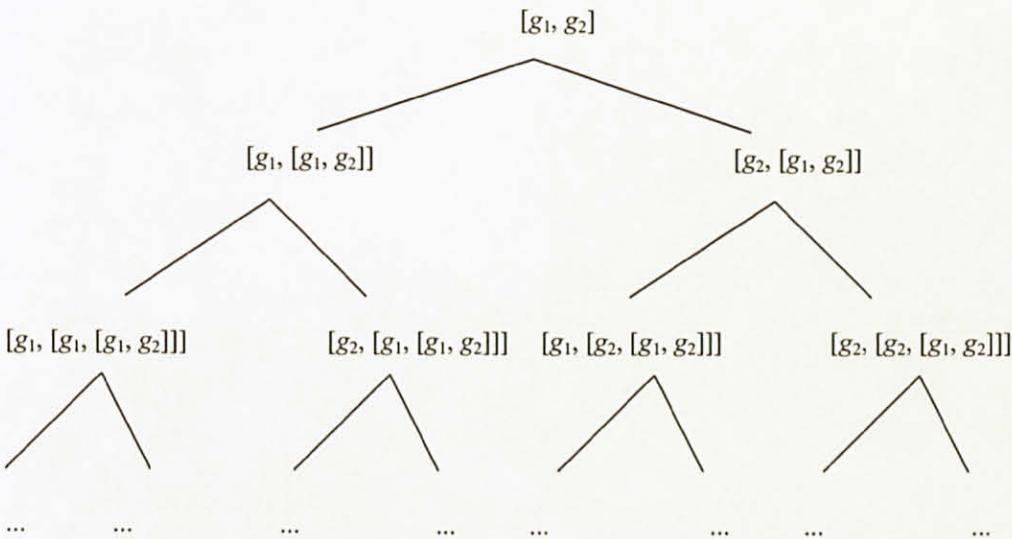


Figure 2.2: Lie Bracket tree

2.2.2 Involutive Distribution and Frobenius Theorem

A distribution, Δ , is the span of all system vector fields and is expressed as

$$\Delta = \text{span}\{g_i(x) : i = 1, \dots, m\} \quad (2.7)$$

A distribution is said to be involutive if it is closed under the Lie Bracket operation. It means every Lie Bracket can be expressed as a linear combination of the system vector fields and belongs to Δ . Therefore the Lie Brackets are unable to escape the Δ and generate new directions of motion. An involutive distribution can be expressed as

$$[g_i, g_j] \in \Delta, \quad \forall g_i, g_j \in \Delta \quad (2.8)$$

According to Frobenius theorem, “A system is completely integrable if and only if its distribution is involutive” [14]. It means that a holonomic system is integrable and its distribution is involutive. Similarly, a nonholonomic system is not integrable and its distribution is not involutive.

2.3 Types of Modeling

The mathematical modeling for the collaborative nonholonomic robots can be obtained using the following two models:

- **Dynamic Model**

The dynamic model takes into consideration the actual forces and torques causing the motion. The dynamic properties for the collaborative robots motion are taken into account. The dynamic equations are obtained using Newton’s laws of motion.

- **Kinematic Model**

The kinematic model is the study of motion without consideration of the force and torque. This type of model allows for the decoupling of collaborative robots dynamics from its movement.

The kinematic model will be further used for analysis and control purpose. The kinematic model for the collaborative robotic system is obtained by taking into consideration each individual robot. There are different models of wheeled mobile robots such as unicycle, car-like robots, etc. The unicycle represents the basic fundamental model of wheeled mobile robots. A unicycle robot consists of one front castor wheel and two rear fixed wheels. The unicycle model can be expanded to represent complex wheeled mobile robots. In this thesis, the unicycle model will be considered for collaborative robots. A unicycle robot can be represented into two orthogonal coordinate systems which are as follows.

- **Global Frame of Reference**

The global or fixed frame coordinates are denoted by (X_G, Y_G) . This frame of reference remains fixed with the origin $(0, 0)$ as shown in Figure 2.3.

- **Local Frame of Reference**

The local or body frame coordinates are denoted by (X_R, Y_R) . This frame of reference remains fixed on the body of the robot with origin at point, p , as shown in Figure 2.3.

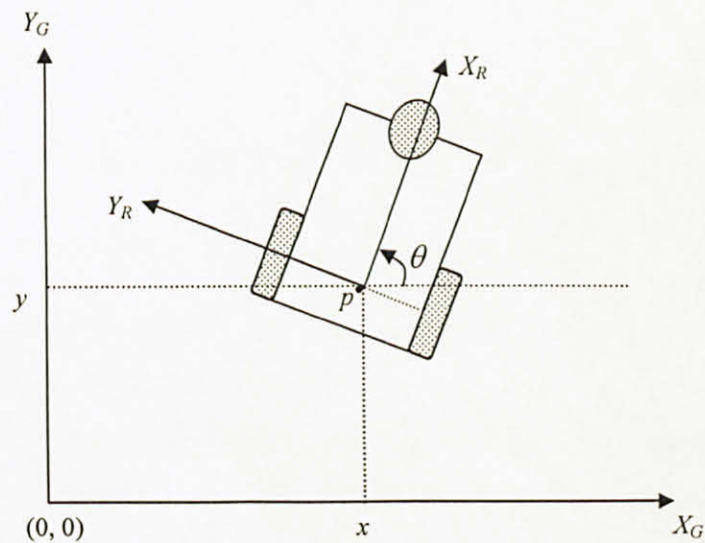


Figure 2.3: Local and global frames for unicycle robot positioning

A robot position can be represented by the vector p which is expressed as

$$p = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad (2.9)$$

where x and y represent the position coordinates with respect to the global x -axis (X_G) and y -axis (Y_G), respectively and θ represents the counterclockwise orientation angle between the robot axle and the global x -axis (X_G). The velocity vector, \dot{p} , can be expressed as

$$\dot{p} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} \quad (2.10)$$

The relationship between the global and local coordinates is given as

$$\dot{p}_R = R(\theta)\dot{p} = R(\theta)[\dot{x} \ \dot{y} \ \dot{\theta}]^T \quad \text{where} \quad (2.11)$$

$$R(\theta) = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where \dot{p}_R is the velocity vector expressed in the local coordinate frame and $R(\theta)$ is the orthogonal rotation matrix.

2.4 Kinematic Modeling of Collaborative Nonholonomic Robots

A collaborative robot system can be described by its state, X , which is a composition of the states of all the robots given as

$$X = [x_1, x_2, \dots, x_n]^T, \quad \dot{X} = F(X, t) \quad (2.12)$$

The state of each robot varies as a function of its continuous state, x_i , and the input vector, u_i . Also each robot receives information about the position of other robots, \hat{z} . The input vector, u_i , depends on the discrete state of the robot, h , which can be either the leader or follower state. The state equations for each robot can be written as

$$\begin{aligned}\dot{x}_i &= f(x_i, u_i) \\ \dot{u}_i &= g_h(x_i, \hat{z})\end{aligned}\tag{2.13}$$

To model the kinematics of each robot in the 2D plane, the configuration $p=[x, y, \theta]^T$ is used. This configuration of the robot stands for three DOF. The kinematic equations for each robot can be written, like the system expressed by Eq. 2.4, as

$$\dot{p} = g_1(p) u_1 + g_2(p) u_2\tag{2.14}$$

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos \theta \\ \sin \theta \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u_2\tag{2.15}$$

where $[u_1, u_2]^T$ are the control inputs. The system modeled by Eq. 2.15 has two system vector fields. One vector allows pure translation, and the other allows pure rotation. In terms of matrices, Eq. 2.15 can be expressed in terms of matrices as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}\tag{2.16}$$

The control inputs $[u_1, u_2]^T$ depend on the discrete state, h , of the robot which can be either leader or follower. The control laws for the leader and follower robots are:

$$\text{Leader} = \begin{cases} u_1 = v_l(t) \\ u_2 = \omega_l(t) \end{cases}, \quad \text{Follower} = \begin{cases} u_1 = v_f(t) \\ u_2 = \omega_f(t) \end{cases}\tag{2.17}$$

where $v(t)$ and $\omega(t)$ represent the translational velocity and angular velocity respectively, the subscript l and f denote the leader and follower robots respectively. The leader-follower formation will be discussed in detail in the next section.

The nonholonomic constraint for wheeled mobile robots assumes that the robots exhibit purely rolling motion and no slipping occur [20], [21]. The nonholonomic constraint is expressed as

$$-\dot{x} \sin \theta + \dot{y} \cos \theta = 0 \quad (2.18)$$

To verify that the kinematic model of Eq. 2.15 is nonholonomic, Frobenius theorem is applied. Comparing the collaborative system model obtained in Eq. 2.15 with the standard driftless system in Eq. 2.4, the following is obtained

$$g_1 = \begin{pmatrix} \cos \theta \\ \sin \theta \\ 0 \end{pmatrix}, \quad g_2 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad (2.19)$$

From Eq. 2.7, Δ of the system vector fields is expressed as

$$\Delta = \text{span}\{g_1, g_2\} \quad (2.20)$$

The Lie Bracket, $[g_1, g_2]$, is given by Eq. 2.21. The Lie Bracket expressed by Eq. 2.21 can be verified using the program `Lie_Bracket.m` (attached in appendix A).

$$[g_1, g_2] = \begin{pmatrix} \sin \theta \\ -\cos \theta \\ 0 \end{pmatrix} \quad (2.21)$$

From Eq. 2.21, the Lie Bracket is linearly independent of g_1 and g_2 and does not belong to the distribution Δ . This means that Δ is not closed under the Lie Bracket operation and therefore, Δ is not involutive. Hence, by Frobenius theorem it is proved that the system modeled by Eq. 2.15 is nonholonomic.

2.5 Leader-Follower Formation for Collaborative Nonholonomic Robots

The control of collaborative robotic system requires coordination at different levels. At the lowest level, it is necessary for each robot to control its motion and to avoid collisions with its neighbors. Furthermore, the robot should move along a desired trajectory. At an immediate supervisory level, it is necessary to maintain a certain formation strategy.

The various approaches to formation control can be divided roughly into three categories: behavior-based, virtual structure formation and leader-follower formation. The behavior-based formation is a distributed approach and has explicit information feedback between neighbors [10]. The virtual structure formation is a centralized approach [11]. Majority of the current algorithms that focus on behavior-based or virtual structure formation are implemented on robots having visual capabilities [12], [13]. Similarly, behavior-based formation focuses on peer to peer communication, whereas in this thesis Bluetooth is considered, which is master-slave architecture [26]. Therefore, leader-follower formation is used for the collaborative robots [22].

In the leader-follower formation, one of the robots is designated as the leader and the others as followers. The leader robot plans and follows a desired trajectory. The follower robots follow the leader robot with a desired distance. The leader robot is responsible for guiding the formation. The leader-follower model comprises of two formation controllers which are as follows.

2.5.1 Separation Bearing Controller (SBC)

The separation bearing controller is used for two robots. The follower robot follows the leader while maintaining a desired relative distance and separation bearing angle with respect to the leader robot. Such type of leader-follower formation control strategy is also denoted by $l - \phi$. A schematic for this control strategy is shown in Figure 2.4. Let ϕ_{lf} denote the separation bearing angle between the leader and follower robot. The separation distance between the center of the leader and the front castor of the follower robot is denoted by l_{lf} . The position coordinates for the front castor of the follower robot is denoted by (x_2, y_2) . The distance between the front castor and the center of axis between the rear wheels for each robot is denoted by d . The leader robot position is expressed by $p_l = [x_l, y_l, \theta_l]^T$ and the control inputs by $u_l = [v_l, \omega_l]^T$. The follower robot position is represented by $p_f = [x_f, y_f, \theta_f]^T$ and the control inputs by $u_f = [v_f, \omega_f]^T$.

The kinematic equations for the leader and follower robot are expressed by Eq. 2.15. Knowing the leader robot position and the separation distance between the leader and follower, the follower robot position can be calculated as given in Eq. 2.22.

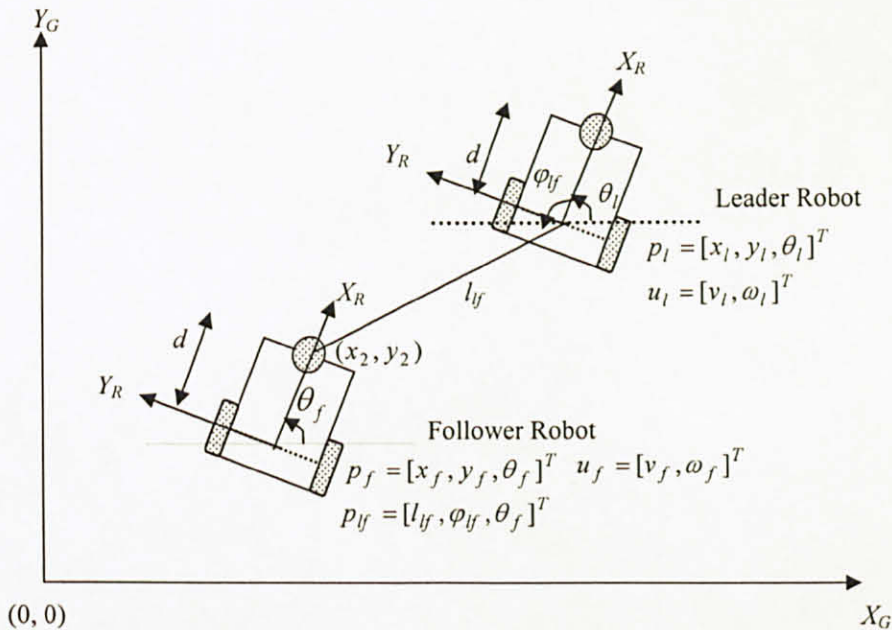


Figure 2.4: Leader-follower formation using separation bearing controller

$$\begin{aligned}x_f &= x_2 + d \cos \theta_f \\y_f &= y_2 + d \sin \theta_f\end{aligned}\tag{2.22}$$

The follower robot can be modeled relatively to the leader robot as $p_{lf} = [l_{lf}, \varphi_{lf}, \theta_f]^T$. The new state vector, p_{lf} , can be expressed through a transformation as $p_{lf} = T_{SB}(p_l, p_f)$ given by

$$\begin{aligned}l_{lf} &= \sqrt{(x_l - x_f - d \cos \theta_f)^2 + (y_l - y_f - d \sin \theta_f)^2} \\ \varphi_{lf} &= \pi - \arctan 2(y_f + d \sin \theta_f - y_l, x_l - x_f - d \cos \theta_f) - \theta_l\end{aligned}\tag{2.23}$$

The original state vector can be recovered through the inverse transformation $p_f = T_{SB}^{-1}(p_l, p_{lf})$. Differentiating Eq. 2.23 and combining with Eq. 2.15, the follower robot kinematic model is obtained as given by Eq. 2.24.

$$\begin{aligned}\dot{l}_{lf} &= v_f \cos \gamma - v_l \cos \varphi_{lf} + d \omega_f \sin \gamma \\ \dot{\varphi}_{lf} &= \frac{v_l \sin \varphi_{lf} - v_f \sin \gamma - \omega_l l_{lf} + d \omega_f \cos \gamma}{l_{lf}} \\ \dot{\theta}_f &= \omega_f\end{aligned}\tag{2.24}$$

where $\gamma = \theta_l - \theta_f + \varphi_{lf}$. In order to avoid collision between the leader and the follower robots, a requirement that $l_{lf} > 2d$ must be ensured. Let $z_{lf} = (l_{lf}, \varphi_{lf})$, so the kinematic system of Eq. 2.24 can be written in compact form as

$$\begin{aligned}\dot{z}_{lf} &= G_{SB}(z_{lf}, \gamma)u_f + F_{SB}(z_{lf})u_l \\ \dot{\theta}_f &= \omega_f\end{aligned}\tag{2.25}$$

where $u_f = (v_f, \omega_f)$, $u_l = (v_l, \omega_l)$ and

$$\begin{aligned}
G_{SB} &= \begin{bmatrix} \cos \gamma & d \sin \gamma \\ -\sin \gamma & d \cos \gamma \\ \frac{1}{l_f} & \frac{1}{l_f} \end{bmatrix} \\
F_{SB} &= \begin{bmatrix} -\cos \varphi_f & 0 \\ \sin \varphi_f / l_f & -1 \end{bmatrix}
\end{aligned} \tag{2.26}$$

The separation bearing controller can be extended to multiple robots when they are marching in a straight line.

2.6.3 Separation-separation Controller (SSC)

This controller is used when multiple robots are present in the formation. Such type of leader-follower formation control strategy is also denoted by $l-l$. A schematic for this control strategy is shown in Figure 2.5. In the $l-l$ formation strategy, the leader robot 2 is actually a follower relative to leader robot 1. The leader robot 2 can be modeled using $l-\varphi$ control strategy. The follower robot can be expressed relative to the leader robot 1 and 2 as $p_f = [l_{1f}, l_{2f}, \theta_f]^T$. In the $l-l$ control strategy, the aim is to maintain the desired lengths l_{1f}^d and l_{2f}^d with respect to both leader robots. Again, to avoid collision $l_{1f} > 2d$ and $l_{2f} > 2d$ must be ensured. The separation distances for the leader robots can be expressed as

$$\begin{aligned}
l_{1f} &= \sqrt{(x_1 - x_f - d \cos \theta_f)^2 + (y_1 - y_f - d \sin \theta_f)^2} \\
l_{2f} &= \sqrt{(x_2 - x_f - d \cos \theta_f)^2 + (y_2 - y_f - d \sin \theta_f)^2}
\end{aligned} \tag{2.27}$$

Differentiating Eq. 2.27, the follower robot kinematics are obtained as

$$\begin{aligned}
\dot{l}_{1f} &= v_f \cos \gamma_1 - v_1 \cos \varphi_{1f} + d \omega_f \sin \gamma_1 \\
\dot{l}_{2f} &= v_f \cos \gamma_2 - v_2 \cos \varphi_{2f} + d \omega_f \sin \gamma_2 \\
\dot{\theta}_f &= \omega_f
\end{aligned} \tag{2.28}$$

where $\gamma_1 = \theta_1 - \theta_f + \varphi_{1f}$ and $\gamma_2 = \theta_2 - \theta_f + \varphi_{2f}$.

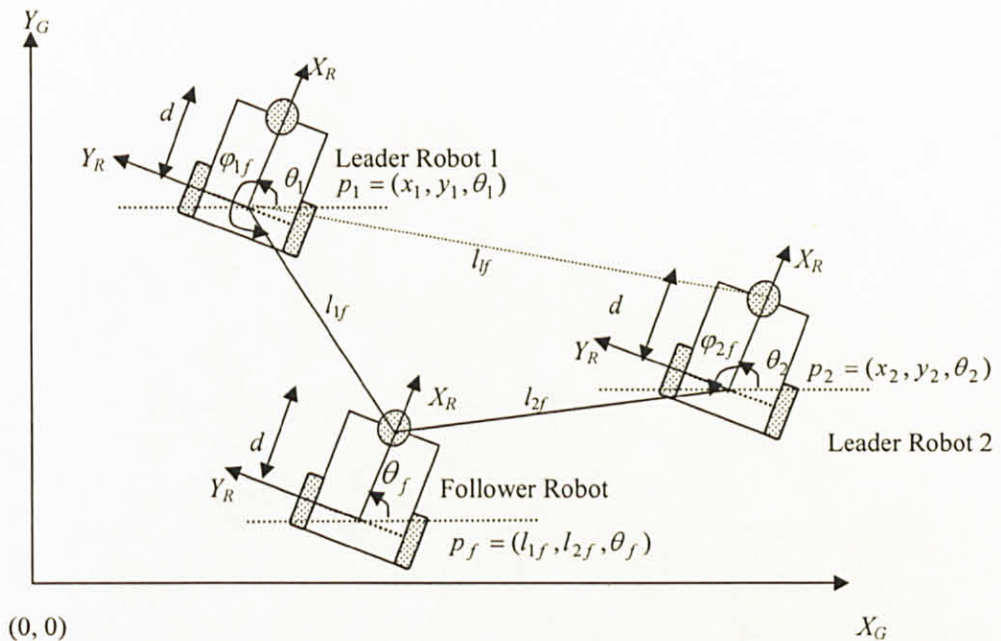


Figure 2.5 : Leader-follower formation using separation-separation controller

Let $z_{lf} = (l_{1f}, l_{2f})$ and $u_f = (v_f, \omega_f)$, so the kinematic equations can be expressed in compact form as

$$\begin{aligned} \dot{z}_{lf} &= \begin{bmatrix} \cos \gamma_1 & d \sin \gamma_1 \\ \cos \gamma_2 & d \sin \gamma_2 \end{bmatrix} u_f + \begin{bmatrix} -v_1 \cos \varphi_{1f} \\ -v_2 \cos \varphi_{2f} \end{bmatrix} \\ \dot{\theta}_f &= \omega_f \end{aligned} \quad (2.29)$$

2.6 Information Sharing among Collaborative Robots

Collaborative robots need to share information while maintaining the leader-follower formation. For information sharing, various communication protocols are available such as Bluetooth, Infrared (Ir), ZigBee, Wireless Fidelity (Wi-Fi) and Ultra-wideband (UWB). A comparative assessment of all these wireless technologies is given in Table 1.

Table 2.1 : Comparison of various wireless technologies

Technologies Properties	Bluetooth	Ir	ZigBee	Wi-Fi	UWB
Cost	Low	Low	Low	High	High at the moment
Market viability	Popular	Popular	Emerging	Popular	Possible Future
Data Transfer Rate	2.1 Mbit/s	4 Mb/s	250 kb/s	54 Mb/s	Upto 100 Mb/s
Power Requirements	1mW	Low	Low	200mW	Low
Effective Range	10-100 m	Up to 2 meters (Line of sight)	10-75m	Up to 31 m	Short (30 to 40 feet)

Based on the properties listed in Table 2.1, the Bluetooth protocol was selected as an information sharing medium among the collaborative robots. Bluetooth is a short-range wireless technology that operates in the license free Industrial, Scientific and Medical (ISM) band at 2.4 GHz [26], [27]. To avoid interference with other devices that uses 2.4 GHz band, the Bluetooth protocol divides the band into 79 channels (each 1 MHz wide). It changes channels up to 1600 times per second. The earlier versions of Bluetooth protocol supported data transfer rate at 723.1 kbit/s. The new version 2.0 supports data transfer rate up to 2.1 Mbits/s.

A Bluetooth profile is a standard interface between Bluetooth devices. Bluetooth profiles are general behaviors through which Bluetooth enabled devices communicate with other devices. There are several profiles available in the Bluetooth protocol suite for communication among devices. One such profile is the Personal Area Network (PAN) or piconet. A piconet can support at the maximum of eight devices in a master-slave relationship. The first Bluetooth device in the piconet is the master and the remaining devices are slaves that communicate with the master. Each device in the piconet is called

as Personal Area Network User (PANU). A piconet defines three modes or roles in which devices can interact with each other. These modes are explained as follows.

- **PANU – PANU**

This mode supports at the maximum two devices. One device acts as a master device and the second one acts as a slave device. Figure 2.6 shows two robots communicating using the PANU – PANU mode.

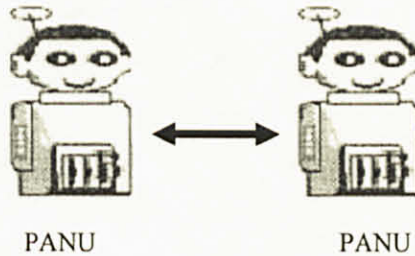


Figure 2.6 : PANU – PANU mode for communication among two robots

- **Group Adhoc Network**

The Group Adhoc Network (GN) enables two or more PAN Users to communicate with each other. The GN device acts as a master and supports at the maximum of seven slaves. The GN mode for communication is shown in Figure 2.7.

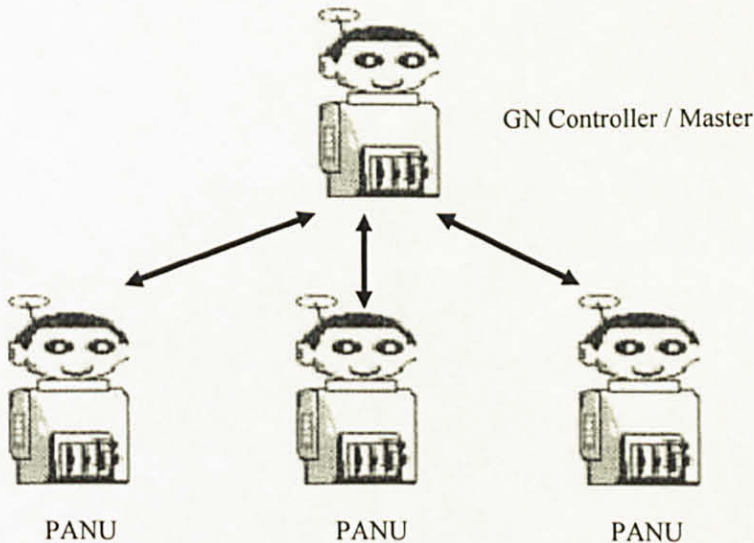


Figure 2.7 : GN mode for communication among robots

- **Network Access Point**

A Network Access Point (NAP) is a Bluetooth device that provides the service of routing network packets. A NAP can act as a bridge between Bluetooth networks and other networks such as Local Area Network (LAN). Figure 2.8 shows the NAP mode for communication among robots.

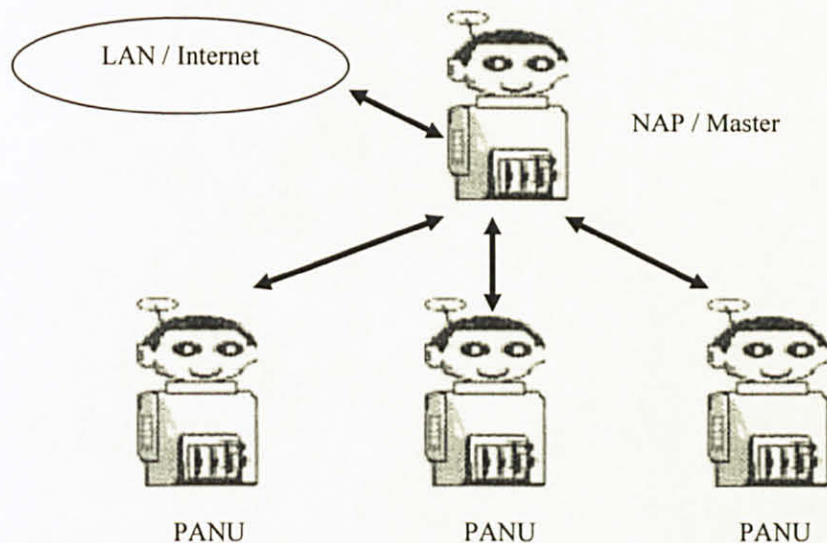


Figure 2.8 : NAP mode for communication among robots

For information sharing among the robots, the GN mode in the piconet is used for communication. The Bluetooth piconet protocol suite is implemented in software as well as in hardware. For simulation of leader-follower formation of robots, Bluetooth USB dongles are used to configure the piconet. A USB dongle is connected to a computer. These dongles are configured to form a Bluetooth piconet using the GN mode. A MATLAB/Simulink session runs on each computer. Each MATLAB/Simulink session communicates with other MATLAB/Simulink sessions in the Bluetooth piconet using the USB dongles. Each session models the leader-follower formation control of robots. The master computer in the piconet models the leader robot in the leader-follower formation. Currently, Bluetooth software protocol suite for PAN profile is commercially provided by Extended Systems (XTND), Microsoft Windows XP Service Pack 2, IVT BlueSoleil and

Widcomm. The summary and limitations of all these software protocol suites are listed in Table 2.2. Based on the properties listed in Table 2.2, IVT BlueSoleil USB dongles are used to form the Bluetooth piconet.

Table 2.2 : Comparison between various protocol suites providers

Protocol Suite Provider	Support for PANU	Support for GN	Support for NAP
XTND	✓		
Microsoft Windows XP Service Pack 2	✓		
IVT BlueSoleil	✓	✓	✓
Widcomm	✓	✓	✓

Currently, the TCPIP toolbox available in MATLAB does not provide support for TCP connections between computers. Rather the TCPIP toolbox provides functions that can be used to acquire data from a network device such as an oscilloscope. So to overcome this limitation, a shared library is needed to be developed and compiled. Therefore, a shared library was developed using Windows Socket programming [28]. This shared library contained functions for message transmission between the computers. This shared library was then compiled using the MATLAB compiler. The format of the message used for communication conforms to the standard Agent Control Language, (ACL), provided by Foundation for Intelligent Physical Agents, (FIPA), [29].

2.7 Summary

In this chapter, the kinematic models for the leader and the follower robots are derived. The follower robots maintain two formation control strategies including separation bearing and separation-separation. A toolbox for communication among the robots is developed and compiled using MATLAB compiler. In the next chapter, the control properties using the kinematic model for the leader-follower formation will be studied.

CHAPTER 3

ANALYSIS OF CONTROL PROPERTIES

This chapter presents the analysis of control properties for the unicycle collaborative nonholonomic robots system. The control analysis involves checking the controllability, stability and observability of the system. The controllability and stability are checked with respect to a point as well as trajectory. The unicycle model of the collaborative robots is considered for analysis purpose. The system is transformed into chained form. Based on the chained form representation, the flat outputs of the system are identified.

3.1 Controllability and Stability at a point

A linear system is completely controllable (all state variables are controllable) if, given any two points in the state space, p and p_o , there exists admissible control inputs capable of taking the system from p to p_o in finite time. To check whether a system is controllable, the controllability matrix, C_M , is computed, which is expressed as

$$C_M = [B \quad AB \quad A^2B \quad \dots \quad A^{n-1}B] \quad (3.1)$$

For a linear controllable system C_M has a full rank of n , where n is the dimension of the configuration space. If a linear system is controllable, then there exists a feedback gain so that the control law $u = k(p - p_e)$ makes the close loop system asymptotically stable about the equilibrium point p_e and the error goes to zero exponentially. For the kinematic model of the unicycle collaborative robots, the approximate linearization of the system at any equilibrium point p_e will be considered. To recall, the kinematic model of the unicycle collaborative robots is expressed as

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos \theta \\ \sin \theta \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u_2 \quad (3.2)$$

The approximate linearization of the system is given as

$$\dot{p} = \delta\ddot{p} = g_1(p_e)u_1 + g_2(p_e)u_2 \quad \text{where } \delta p = p - p_e \quad (3.3)$$

$$\dot{p} = \delta(\dot{p}) = \begin{bmatrix} \cos \theta_e \\ \sin \theta_e \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \quad (3.4)$$

Eq. 3.4 can be written in terms of matrices as

$$\dot{p} = \delta(\dot{p}) = \begin{bmatrix} \cos \theta_e & 0 \\ \sin \theta_e & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (3.5)$$

The controllability matrix, C_M , is expressed by Eq. 3.6. This matrix can also be verified using the program `Linear_Controlability.m` (attached in Appendix B).

$$C_M = [B \quad AB \quad A^2B] = \begin{bmatrix} \cos \theta_e & 0 & 0 & 0 & 0 & 0 \\ \sin \theta_e & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.6)$$

$$\text{rank}(C_M) = 2$$

From Eq. 3.6, the rank of C_M is less than the order of the system, n , which is 3. This indicates that the linearized system expressed by Eq. 3.5 is not controllable. This implies that a linear controller will never achieve posture stabilization, not even in a local sense. Thus, if the system is linearized at an equilibrium point, the linearized system is not controllable. Hence, a linear control will not work here.

In order to study the controllability of collaborative nonholonomic robots, tools from nonlinear control theory will be used. Since the system modeled by Eq. 3.2 is driftless, the term local accessibility and controllability can be used interchangeably. The controllability of the system can be established using the Chow-Rashevski theorem [30].

According to this theorem, for a driftless control system, if the accessibility rank condition, given by Eq. 3.7, holds, then the control system is locally accessible (controllable) from p_o .

$$\text{rank } \Delta_c(p_o) = n \quad (3.7)$$

where n is the order of the state space system and Δ_c is the accessibility distribution. Δ_c is defined as the span of all the input vector fields and associated Lie Brackets.

$$\begin{aligned} \Delta_c &= \text{span}\{v \mid v \in \Delta_i \ \forall i \geq 1\} \\ \Delta_i &= \Delta_{i-1} + \text{span}\{[g, v] \mid g \in \Delta_1, v \in \Delta_{i-1}\}, i \geq 2 \\ \Delta_1 &= \text{span}\{g_1, g_2, \dots, g_m\} \end{aligned} \quad (3.8)$$

For the kinematic model given by Eq. 3.2, the accessibility distribution Δ_c is computed as follows.

$$\begin{aligned} \Delta_c &= [g_1 \quad g_2 \quad [g_1, g_2]] \\ \Delta_c &= \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ \sin \theta & 0 & -\cos \theta \\ 0 & 1 & 0 \end{bmatrix} \\ \text{rank } (\Delta_c) &= 3 = n \end{aligned} \quad (3.9)$$

As the rank of Δ_c is 3, which is equal to the order of the configuration space, therefore, the kinematic model of Eq. 3.2 is nonlinearly controllable at a point. Controllability can also be shown constructively, i.e., by providing an explicit sequence of maneuvers bringing the robots from any start configuration (x_s, y_s, θ_s) to any desired goal configuration (x_g, y_g, θ_g) . Since each robot can rotate itself, this task is simply achieved by an initial rotation on (x_s, y_s) until the robot is oriented towards (x_g, y_g) , followed by a translation to the goal position, and by a final rotation on (x_g, y_g) so as to align θ with θ_g .

Regarding the stability of the system at a point, Lypaunov (asymptotic) stability cannot be achieved by means of a smooth, time invariant feedback. This result is established on the basis of Brockett's theorem [6] which states that the stabilization of a driftless regular system (a system in which the input vector fields are well defined and linearly independent at p_e) by a smooth time invariant feedback is not possible. It further implies the number of inputs, m , should be equal to the number of states, n , as both a necessary and sufficient condition for smooth stabilization.

To obtain posture stabilization, it is obligatory to give up the continuity requirement, i.e. to include the non smooth (discontinuous) feedback or to apply the time varying control laws or to apply a combination of both.

3.2 Controllability and Stability around a Trajectory

To check the controllability of the collaborative robots around a trajectory, a desired trajectory and inputs are considered as $p_d(t) = [x_d(t), y_d(t), \theta_d(t)]^T$ and $u_d(t) = [v_d(t), \omega_d(t)]^T$, respectively. In order to be feasible, the desired trajectory must satisfy the nonholonomic constraint for the robots. Defining the state tracking error as $\tilde{p}(t) = p(t) - p_d(t)$ and the input variations as $\tilde{v}(t) = v(t) - v_d(t)$ and $\tilde{\omega}(t) = \omega(t) - \omega_d(t)$, the approximate linearization of the unicycle system about the reference trajectory is obtained as

$$\dot{\tilde{p}} = A(t)\tilde{p} + B(t) \begin{bmatrix} \tilde{v} \\ \tilde{\omega} \end{bmatrix} \text{ where} \quad (3.10)$$

$$A(t) = \sum_{i=1}^m \frac{\partial g_i}{\partial p} \bigg|_{p=p_d} \cdot u_{di}(t) \quad , \quad B(t) = G(p_d(t)) \quad (3.11)$$

For the kinematic model of Eq. 3.2, the following is obtained.

$$A(t) = v_d \frac{\partial g_1}{\partial p} \bigg|_{p=p_d} + \omega_d \frac{\partial g_2}{\partial p} \bigg|_{p=p_d} \quad , \quad B(t) = G(p_d(t)) \quad (3.12)$$

Computing the values for $A(t)$ and $B(t)$, Eq 3.13 is obtained. This can also be verified using the program `Linearized_A.m` (attached in Appendix C).

$$A(t) = \begin{bmatrix} 0 & 0 & -v_d \sin \theta_d(t) \\ 0 & 0 & v_d \cos \theta_d(t) \\ 0 & 0 & 0 \end{bmatrix}, \quad B(t) = \begin{bmatrix} \cos \theta_d(t) & 0 \\ \sin \theta_d(t) & 0 \\ 0 & 1 \end{bmatrix} \quad (3.13)$$

Substituting the values in Eq. 3.10, the approximate linearized system is expressed as

$$\dot{\tilde{p}} = \begin{bmatrix} 0 & 0 & -v_d \sin \theta_d(t) \\ 0 & 0 & v_d \cos \theta_d(t) \\ 0 & 0 & 0 \end{bmatrix} \tilde{p} + \begin{bmatrix} \cos \theta_d(t) & 0 \\ \sin \theta_d(t) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{v} \\ \tilde{\omega} \end{bmatrix} \quad (3.14)$$

Since the linearized system of Eq. 3.14 is time varying, the controllability analysis involves to check whether the controllability matrix C_M is nonsingular. However a simpler analysis can be performed by transforming the state tracking error, \tilde{p} , into the local or body frame coordinates, expressed by \tilde{p}_R in Eq. 2.11, as

$$\tilde{p}_R = \begin{bmatrix} \cos \theta_d & \sin \theta_d & 0 \\ -\sin \theta_d & \cos \theta_d & 0 \\ 0 & 0 & 1 \end{bmatrix} \tilde{p} \quad (3.15)$$

Next, similarity transformation is performed to change from global frame to the local frame coordinates (see appendix D for proof). Let T denote the transformation matrix, so using the similarity transformation, the system can be expressed as

$$\dot{\tilde{p}}_R = (\dot{T} T^{-1} + T A T^{-1}) \tilde{p}_R + (T B) \tilde{u} \quad (3.16)$$

The similarity transformation for the system of Eq. 3.14 is given by Eq. 3.17. This transformation can also be verified using the program `Similarity_Transform.m` (attached in Appendix E).

$$\dot{\tilde{p}}_R = \begin{bmatrix} 0 & \omega_d & 0 \\ -\omega_d & 0 & v_d \\ 0 & 0 & 0 \end{bmatrix} \tilde{p}_R + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{v} \\ \tilde{\omega} \end{bmatrix} \quad (3.17)$$

When v_d and ω_d are constant, the above linear system becomes time-invariant hence the controllability can be checked using the controllability matrix, C_M . The controllability matrix, C_M , is given by Eq. 3.18. This matrix can also be verified using the program `Linear_Controlability.m` (attached in Appendix B).

$$C_M = [B \quad AB \quad A^2B] = \begin{bmatrix} 1 & 0 & 0 & 0 & -\omega_d^2 & v_d\omega_d \\ 0 & 0 & -\omega_d & v_d & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.18)$$

$\text{rank}(C_M) = 3$

From Eq. 3.18, C_M has a rank of 3 provided that either v_d or ω_d is nonzero. Therefore, the conclusion is that the kinematic system of Eq. 3.2 can be locally stabilized around a reference trajectory by linear feedback.

3.3 Observability

A system is completely observable if the state vector can be determined in finite time using only the input, $u(t)$ and/or output, $o(t)$ of the system. Let p denote the state vector and o denote the output vector given as

$$o = h(p) \quad (3.19)$$

Let $L_f h(p)$ denote the Lie derivative of the vector field f along h at point p , which is expressed as

$$L_f h(p) = \nabla_f h(p) \quad (3.20)$$

The observability map, $\Phi(p)$, for a system is expressed as

$$\Phi(p) = \begin{bmatrix} h(p) \\ L_f h(p) \\ L_f^2 h(p) \\ \vdots \\ L_f^{n-1} h(p) \end{bmatrix} \quad \text{with} \quad \frac{\partial}{\partial p} \Phi(p) = \begin{bmatrix} dh(p) \\ dL_f h(p) \\ dL_f^2 h(p) \\ \vdots \\ dL_f^{n-1} h(p) \end{bmatrix} \quad (3.21)$$

The Jacobian of observability map is called the observability matrix and is denoted by $\frac{\partial}{\partial p} \Phi(p)$. For a nonlinear system to be observable, the observability matrix must be of

full rank. For the kinematic model of Eq. 3.2, the output can be chosen as

$$o = h(p) = C.p = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad (3.22)$$

This results in the following observability map

$$\Phi(p) = \begin{bmatrix} o \\ \dot{o} \\ \ddot{o} \end{bmatrix} = \begin{bmatrix} h(p) \\ L_f h(p) \\ L_f^2 h(p) \end{bmatrix} \quad (3.23)$$

Substituting the values, the observability map is written as

$$\Phi(p) = \begin{bmatrix} x \\ y \\ v \cos \theta \\ v \sin \theta \\ -v\omega \sin \theta \\ v\omega \cos \theta \end{bmatrix} \quad (3.24)$$

The observability matrix can be computed as

$$\frac{\partial}{\partial p} \Phi(p) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -v \sin \theta \\ 0 & 0 & v \cos \theta \\ 0 & 0 & -v\omega \cos \theta \\ 0 & 0 & v\omega \sin \theta \end{bmatrix} \quad (3.25)$$

The observability matrix has full rank and is well defined for all values of θ . As long as $v \neq 0$, this rank is preserved for all θ . Therefore the unicycle model for collaborative robotic system is observable. For the leader-follower formation, the output vector for the i^{th} follower robot, using the SBC controller expressed by Eq. 2.25, can be chosen as

$$o = C.p = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} l_f \\ \varphi_f \\ \theta_f \end{bmatrix} \quad (3.26)$$

From Eq. 3.26 $[o_1, o_2]^T = [\varphi_f, \theta_f]^T$, the observability matrix can be written as

$$\frac{\partial}{\partial p} \Phi(p) = \begin{bmatrix} \frac{\partial}{\partial l_f} o_1 & \frac{\partial}{\partial \varphi_f} o_1 & \frac{\partial}{\partial \theta_f} o_1 \\ \frac{\partial}{\partial l_f} o_2 & \frac{\partial}{\partial \varphi_f} o_2 & \frac{\partial}{\partial \theta_f} o_2 \\ \frac{\partial}{\partial l_f} \dot{o}_1 & \frac{\partial}{\partial \varphi_f} \dot{o}_1 & \frac{\partial}{\partial \theta_f} \dot{o}_1 \\ \frac{\partial}{\partial l_f} \dot{o}_2 & \frac{\partial}{\partial \varphi_f} \dot{o}_2 & \frac{\partial}{\partial \theta_f} \dot{o}_2 \\ \frac{\partial}{\partial l_f} \ddot{o}_1 & \frac{\partial}{\partial \varphi_f} \ddot{o}_1 & \frac{\partial}{\partial \theta_f} \ddot{o}_1 \\ \frac{\partial}{\partial l_f} \ddot{o}_2 & \frac{\partial}{\partial \varphi_f} \ddot{o}_2 & \frac{\partial}{\partial \theta_f} \ddot{o}_2 \end{bmatrix} \quad (3.27)$$

The determinant of the observability matrix is expressed as

$$\det\left(\frac{\partial}{\partial p}\Phi(p)\right) = -\frac{\partial\dot{\phi}_f}{\partial l_f} = \frac{1}{l_f}[\dot{\phi}_f + \omega_l] \quad (3.28)$$

Therefore, if $\det\left(\frac{\partial}{\partial p}\Phi(p)\right) \neq 0$, the states of the follower robot are observable. Hence the unicycle as well as the leader-follower model for collaborative nonholonomic robots is observable.

3.4 Chained Form

Chained forms are canonical model structures for the development of both open-loop and closed-loop control strategies for nonholonomic systems [30]. Canonical model structures can be categorized into three categories which are as follows.

- Chained form
- Power form
- Caplygin form

In case of collaborative nonholonomic robots modeled by Eq. 3.2, the above three forms are equivalent via a coordinate transformation [31]. The chained forms were first introduced in [32]. A two-input driftless control system having n order, can be expressed by $(2, n)$ chained form as

$$\begin{aligned} \dot{z}_1 &= v_1 \\ \dot{z}_2 &= v_2 \\ \dot{z}_3 &= z_2 v_1 \\ &\vdots \\ \dot{z}_n &= z_{n-1} v_1 \end{aligned} \quad (3.29)$$

where z_1, z_2 are called the base variables and v_1, v_2 are the generating inputs. The $(2, n)$ chained form can be shown to be completely controllable using the Chow-Rashevski Lie algebra rank condition. In performing this calculation, all Lie Brackets above the order of $n-2$ are identically zero; this property of the system is called as nilpotency. The chained forms conversion for a system may not be unique. For the unicycle collaborative robot system modeled by Eq. 3.2, the following change of coordinates and inputs is introduced.

$$\begin{aligned}
 z_1 &= x \\
 z_2 &= \tan \theta \\
 z_3 &= y \\
 u_1 &= v_1 / \cos \theta \\
 u_2 &= v_2 \cos^2 \theta
 \end{aligned} \tag{3.30}$$

The chained form for the system is expressed as

$$\begin{aligned}
 \dot{z}_1 &= v_1 \\
 \dot{z}_2 &= v_2 \\
 \dot{z}_3 &= z_2 v_1
 \end{aligned} \tag{3.31}$$

The conversion to chained form using Eq. 3.30 is not unique. Another possibility for conversion to chained form is given by Eq. 3.32.

$$\begin{aligned}
 z_1 &= \theta \\
 z_2 &= x \cos \theta + y \sin \theta \\
 z_3 &= x \sin \theta - y \cos \theta \\
 u_1 &= z_3 v_1 + v_2 \\
 u_2 &= v_2
 \end{aligned} \tag{3.32}$$

The chained form will be further used in the design of feedback control strategies for collaborative robots. The chained forms are also useful in determining the flat outputs for the system.

3.5 Differential Flatness

A nonlinear system is differentially flat if a set of variables, called flat outputs, can be found such that all the inputs, states and outputs of the system can be determined algebraically from them without integration. The algebraic expressions may involve the flat outputs and a finite number of its high order derivatives. The flat outputs are denoted by o , the states by z and inputs by u , such that

$$\begin{aligned} z &= f_1(o, \dot{o}, \ddot{o}, \dots, o^{[r]}) \\ u &= f_2(o, \dot{o}, \ddot{o}, \dots, o^{[r]}) \end{aligned} \quad (3.33)$$

where r denotes a certain number of derivative, f_1 and f_2 denote some algebraic functions. For a driftless system, flatness is equal to chained-form transformability. The flat outputs of the chained form are z_1, z_n . For the collaborative robot chained form representation using Eq. 3.30, (x, y) are the flat outputs.

Differential flatness is useful in trajectory generation where it reduces the problem of trajectory generation to finding a trajectory of the flat outputs. Once the flat outputs are identified, the remaining states of the system as well as the inputs can be computed using algebraic transformations.

3.6 Summary

In this chapter, the control properties are analyzed for collaborative robots. The robots are linearly not controllable at a point, but nonlinearly controllable. The leader robot model is observable and the follower robot model using separation bearing and separation-separation formation is also observable. The leader robot model is transformed into chained forms and flat outputs which will be useful in generating the feedforward control inputs. In the next chapter, feedback controllers for trajectory tracking and posture stabilization are presented. The stability analysis for the feedback controllers is also discussed.

CHAPTER 4

FEEDBACK LINEARIZATION TECHNIQUES

This chapter presents the feedback controllers for collaborative nonholonomic robots. Feedback controllers can be designed using cascaded system theory, stable tracking method based on linearization of error model, approximate linearization, nonlinear design and feedback linearization. The full state and input-output feedback linearization design techniques are discussed in detail for the leader and follower robots. Furthermore, the stability analysis of the feedback controllers is also discussed. Finally, the posture stabilization control strategies for the leader robot are presented.

4.1 Mathematical Preliminaries

The basic principle in designing feedback control strategies is to ensure the stability of the system. To recall, some basic concepts regarding stability are presented.

4.1.1 Equilibrium Point

A nonautonomous system is written as a function of time. Let p represent the state of a nonautonomous system and t represent the time, so a nonautonomous system can be expressed as

$$\dot{p} = f(p, t) \quad (4.1)$$

A point, $p = p^*$, in the state space is an equilibrium point for the system, if it has the property that whenever the state of the system starts at p^* , it will remain at p^* for all future times. For the system of Eq. 4.1, the equilibrium points can be expressed as

$$f(p, t) = 0 \quad (4.2)$$

Furthermore, an equilibrium point is said to be isolated if there exists a $\delta > 0$ such that for any other equilibrium point \tilde{p}_e , $\|p^* - p_e\| > \delta$.

4.1.2 Types of Continuous Function

A function, $f: \mathbb{R}^n \rightarrow \mathbb{R}^m$, is continuous between two points p_1 and p_2 , if given any arbitrary ε , a constant $\delta > 0$ exists such that Eq. 4.3 is satisfied.

$$\|p_1 - p_2\| < \delta \Rightarrow \|f(p_1) - f(p_2)\| < \varepsilon; \quad p_1, p_2 \in \mathbb{R}^n \quad (4.3)$$

A function is continuous on a set S , if it is continuous at every point in S . The function is piece-wise continuous on S , if it is continuous on S , except for a finite number of points. The function is uniformly continuous on S , if given any arbitrary ε , a constant $\delta > 0$ exists such that Eq. 4.3 holds.

A continuous function, $\alpha: [0, a) \rightarrow [0, \infty)$, to class \mathcal{K} ($\alpha \in \mathcal{K}$) if it is strictly increasing and $\alpha(0) = 0$ [30]. The function, α , belongs to the class \mathcal{K}_∞ if $a = \infty$ and $\alpha(t) \rightarrow \infty$ as $t \rightarrow \infty$.

A continuous function, $\beta: [0, a) \times [0, \infty) \rightarrow [0, \infty)$, belongs to class \mathcal{KL} ($\beta \in \mathcal{KL}$) if for each fixed t , $\beta(s, t)$ belongs to class \mathcal{K} with respect to t and if for each fixed t , $\beta(s, t)$ is decreasing with respect to s and $\beta(s, t) \rightarrow 0$ as $s \rightarrow \infty$.

4.1.3 Stability of Equilibrium Point

The equilibrium point $p^* = 0$ is

- stable if, for each $\varepsilon > 0$, there is $\delta = \delta(\varepsilon) > 0$ such that Eq. 4.4 is satisfied.

$$\|p(t_o)\| < \delta \Rightarrow \|p(t)\| < \varepsilon; \quad \forall t \geq t_o \geq 0 \quad (4.4)$$

- uniformly stable if, for each $\varepsilon > 0$, there is $\delta = \delta(\varepsilon) > 0$, independent of t_o , such that Eq. 4.4 is satisfied.
- asymptotically stable at t_o , if it is stable and there is a positive constant $c = c(t_o)$ such that $p(t) \rightarrow 0$ as $t \rightarrow \infty$ for all $\|p(t_o)\| < c$.
- globally asymptotically stable if it is stable and $\lim_{t \rightarrow \infty} p(t) = 0$ for all $p_o \in \mathfrak{R}^n$.
- locally uniformly asymptotically stable if, there exist a function $\beta \in \mathcal{KL}$ and a positive constant r , such that for all $t > t_o > 0$ and for all initial state $\|p(t_o)\| < r$, Eq. 4.5 is satisfied.

$$\|p(t)\| < \beta(\|p(t_o)\|, t - t_o) \quad (4.5)$$

- globally uniformly asymptotically stable if, Eq. 4.5 is satisfied with $\beta \in \mathcal{KL}_\infty$ for any initial state $p(t_o)$.
- locally exponentially stable if, Eq. 4.5 is satisfied with $\beta(s, t) = kte^{-\gamma s}$ and $k > 0, \gamma > 0$ for $\|p(t_o)\| < s$.
- globally exponentially stable is Eq. 4.5 is satisfied with $\beta(s, t) = kte^{-\gamma s}$ and $k > 0, \gamma > 0$ for any initial state $p(t_o)$.
- globally \mathcal{K} -exponentially stable if a function with $k \in \mathcal{K}$ and a constant $\gamma > 0$ exist such that for all $(t_o, p(t_o)) \in \mathfrak{R}^+ \times \mathfrak{R}$, Eq 4.6 is satisfied.

$$\|p(t)\| \leq k(\|p(t_o)\|), e^{-\gamma(t-t_o)} \quad (4.6)$$

4.1.4 Lyapunov Stability

Let $p = 0$ be the equilibrium point for $\dot{p} = f(p)$ and $D \subset \mathbb{R}^n$ be the domain containing $p = 0$. Let $V : D \rightarrow \mathbb{R}$ be a continuously differentiable function such that Eq. 4.7 holds, then $p = 0$ is stable and $V(p)$ is called a Lyapunov function [14], [30].

$$\begin{aligned} V(0) = 0 \text{ and } V(p) > 0 \text{ in } D \setminus \{0\}; \\ \dot{V}(p) \leq 0 \text{ in } D \end{aligned} \tag{4.7}$$

In addition to Eq. 4.7, if Eq. 4.8 is also satisfied, then $p = 0$ is asymptotically stable.

$$\dot{V}(p) < 0 \text{ in } D \setminus \{0\} \tag{4.8}$$

For a nonautonomous system, $\dot{p} = f(p, t)$, assuming $V : \{0, \infty\} \times D \rightarrow \mathbb{R}$ be a continuously differentiable function. The equilibrium point, $p = 0$, is uniformly asymptotically stable if Eq. 4.9 is satisfied.

$$\begin{aligned} k_1(p) \leq V(p, t) \leq k_2(p), \\ \dot{V}(p, t) = \frac{\partial V}{\partial t} + \frac{\partial V}{\partial p} f(p, t) \leq -k_3(p), \quad \forall t \geq 0 \text{ and } \forall p \in D \end{aligned} \tag{4.9}$$

where $k_1(p)$, $k_2(p)$ and $k_3(p)$ are continuous positive definite functions on D . If $k_3(p) = 0$, then $p = 0$ is uniformly stable. Furthermore, if $k_1(p) \rightarrow \infty$ for all $\|p\| \rightarrow \infty$, then $p = 0$ is globally uniformly asymptotically stable.

If the conditions in Eq. 4.9 are replaced as

$$\begin{aligned} k_1 \|p\|^q &\leq V(p,t) \leq k_2 \|p\|^q, \\ \dot{V}(p,t) &= \frac{\partial V}{\partial t} + \frac{\partial V}{\partial p} f(p,t) \leq -k_3 \|p\|^q. \end{aligned} \quad (4.10)$$

for some positive constants k_1, k_2, k_3 and q , then $p = 0$ is exponentially stable. Furthermore, if the assumptions are satisfied for all $p \in \mathfrak{R}^n$, then $p = 0$ is globally exponentially stable.

4.2 Feedback Control Design Techniques

In automatic control systems, feedback improves the system performance by completing the task even if external disturbances and initial errors are present. Hence the effect of unmodeled events at running time, such as occasional slipping of the wheels or erroneous initial localization, is minimized. Furthermore, feedback control strategies can be used to stabilize the system. There are various design techniques available for feedback control, some of which are stated as follows [33].

- Root locus method
- PID method
- Poles placement
- Cascaded systems theory – can be applied to nonlinear system
- Linearization of corresponding error model – can be applied to nonlinear system
- Approximate linearization – applied to nonlinear system
- Feedback linearization – applied to nonlinear system

The design of nonlinear feedback control systems is a challenging task. A common practice is to linearize the system. A system can be linearized in the following ways.

4.2.1 Taylor Series Expansion

This technique linearizes a nonlinear system around its nominal operating point, c , by expanding the nonlinearity using Taylor series. Mathematically, it can be expressed as transforming the following nonlinear system

$$\dot{p} = f(p, u) \quad (4.11)$$

into a locally linearized system of the form

$$\dot{\tilde{p}} = A\tilde{p} + B\tilde{u} \quad (4.12)$$

where \tilde{p} and \tilde{u} are small deviations given as

$$\begin{aligned} \tilde{p}(t) &= p(t) - p_c(t) \\ \tilde{u}(t) &= u(t) - u_c(t) \end{aligned} \quad (4.13)$$

The coefficients A and B are calculated as

$$A = \left. \frac{\partial f}{\partial x} \right|_{x_c}, \quad B = \left. \frac{\partial f}{\partial u} \right|_{u_c} \quad (4.14)$$

This technique has some limitations such as

- Retaining only the first order term and discarding the higher order terms may result in a highly inaccurate linearized system.
- The resulting linearized system may not be uncontrollable.

Hence, this technique is not suitable for designing feedback control strategies for a nonlinear system.

4.2.2 Feedback Linearization

Feedback linearization is the procedure of algebraically transforming a nonlinear system into a (fully or partly) linear one, so that linear control techniques can be applied. It differs entirely from conventional linearization, such as Jacobian linearization, in that feedback linearization is achieved using exact state transformations and feedback, rather than by linear approximations of the dynamics.

Feedback linearized control strategies are designed based on state and/or input. As the system modeled by Eq. 2.15 is having two inputs, therefore, feedback linearized strategies for Multi Input Multi Output systems, MIMO, are considered. There are two techniques for feedback linearization of a MIMO nonlinear system which are as follows [14], [30].

- **Full State Feedback Linearization**

In full state feedback transformation, the whole set of the system equations become linear. This means that the state equations are completely linearized.

- **Input-output Feedback Linearization**

In input-output feedback transformation, the input and output response of the close loop system is linear. For MIMO systems, this transformation results in the decoupling of input and output vectors. This means that the input output map is linearized, while the state equation may only be partially linearized.

To design controllers either using full state or input output linearization, static and dynamic feedback can be used. For the leader robot, the unicycle model would be used to design feedback control strategies. For the follower robots, the separation bearing and separation-separation formation control models would be feedback linearized.

4.3 Feedback Controller using Approximate Linearization

Let $[x_d, y_d, \theta_d]^T$ represent the desired trajectory for the leader robot. The desired inputs are denoted by $[v_d, \omega_d]^T$. The control objective of the feedback controller is to drive the errors $[x_d - x, y_d - y, \theta_d - \theta]^T$ to zero. To recall, the unicycle model for the leader robot is shown in Figure 4.1. The kinematic equations for the unicycle model of the leader robot are expressed as

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos \theta \\ \sin \theta \\ 0 \end{pmatrix} v + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \omega \quad (4.15)$$

The error, e , is expressed in the moving frame, (X_R, Y_R) , as

$$e = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_d - x \\ y_d - y \\ \theta_d - \theta \end{bmatrix} \quad (4.16)$$

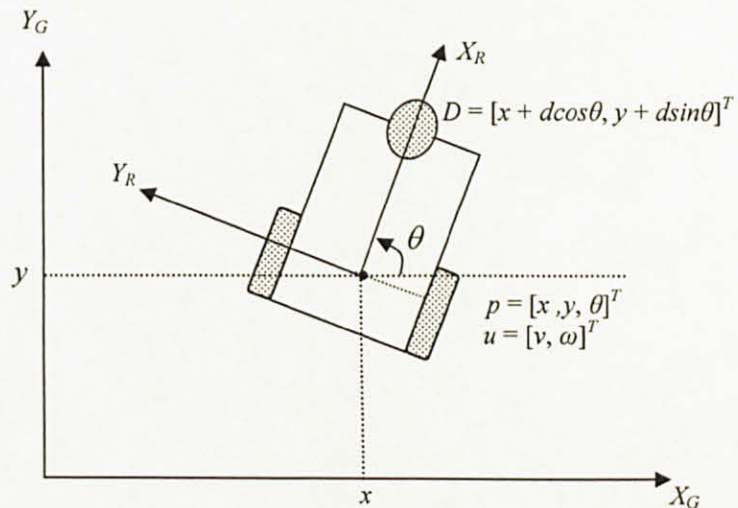


Figure 4.1: Leader robot positioning

Introducing a change of inputs as in [17], [25]

$$\begin{aligned} u_1 &= -v + v_d \cos e_3 \\ u_2 &= \omega_d - \omega \end{aligned} \quad (4.17)$$

Differentiating Eq. 4.16 and combining with Eq. 4.17, the error dynamics are expressed as

$$\dot{e} = \begin{bmatrix} 0 & \omega & 0 \\ -\omega & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} e + \begin{bmatrix} 0 \\ \sin e_3 \\ 0 \end{bmatrix} v_d + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (4.18)$$

Linearizing Eq. 4.18 about the equilibrium point, $e = 0$ and $u = 0$, the following is obtained

$$\dot{e} = \begin{bmatrix} 0 & \omega_d(t) & 0 \\ -\omega_d(t) & 0 & v_d(t) \\ 0 & 0 & 1 \end{bmatrix} e + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (4.19)$$

Eq. 4.19 represents a time varying system. Assuming $v_d(t) = v$ and $\omega_d(t) = \omega$, Eq. 4.19 becomes a linear time invariant system. The feedback law for the system is expressed as

$$\begin{aligned} u_1 &= -k_1 e_1 \\ u_2 &= -k_2 \text{sign}(v_d) e_2 - k_3 e_3 \end{aligned} \quad (4.20)$$

where k_1 , k_2 and k_3 are feedback coefficients. The desired close-loop characteristic equation is

$$(\lambda + 2\zeta a)(\lambda^2 + 2\zeta a\lambda + a^2), \quad \zeta, a > 0 \quad (4.21)$$

Comparing Eq. 4.20 with the desired characteristic equation, the feedback gains are obtained as

$$\begin{aligned} k_1 = k_3 &= 2\zeta a = 2\zeta \sqrt{w_d^2(t) + bv_d^2(t)} \\ k_2 &= b|v_d(t)| \end{aligned} \quad (4.22)$$

Substituting back to obtain the original control inputs, this design leads to the following nonlinear time-varying controller.

$$\begin{aligned} v &= v_d \cos(\theta_d - \theta) + k_1 [\cos\theta(x_d - x) + \sin\theta(y_d - y)] \\ \omega &= \omega_d + k_2 \text{sign}(v_d) [\cos\theta(y_d - y) - \sin\theta(x_d - x)] + k_3(\theta_d - \theta) \end{aligned} \quad (4.23)$$

For the control law of Eq. 4.23, even if the eigenvalues are constant and with negative real part, asymptotic stability is not guaranteed because the system is still time-varying. The leader robot kinematics model is shown in Figure 4.2. The approximate linearized feedback controller is shown in Figure 4.3. The leader robot kinematic model connected to the approximate linearized feedback controller is shown in Figure 4.4.

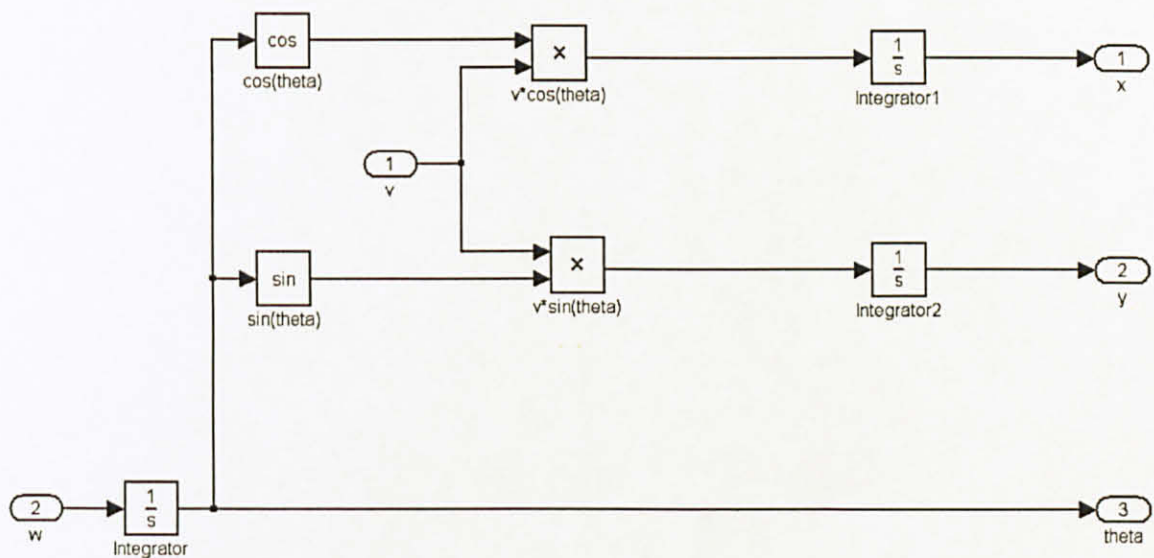


Figure 4.2: Leader robot kinematic model

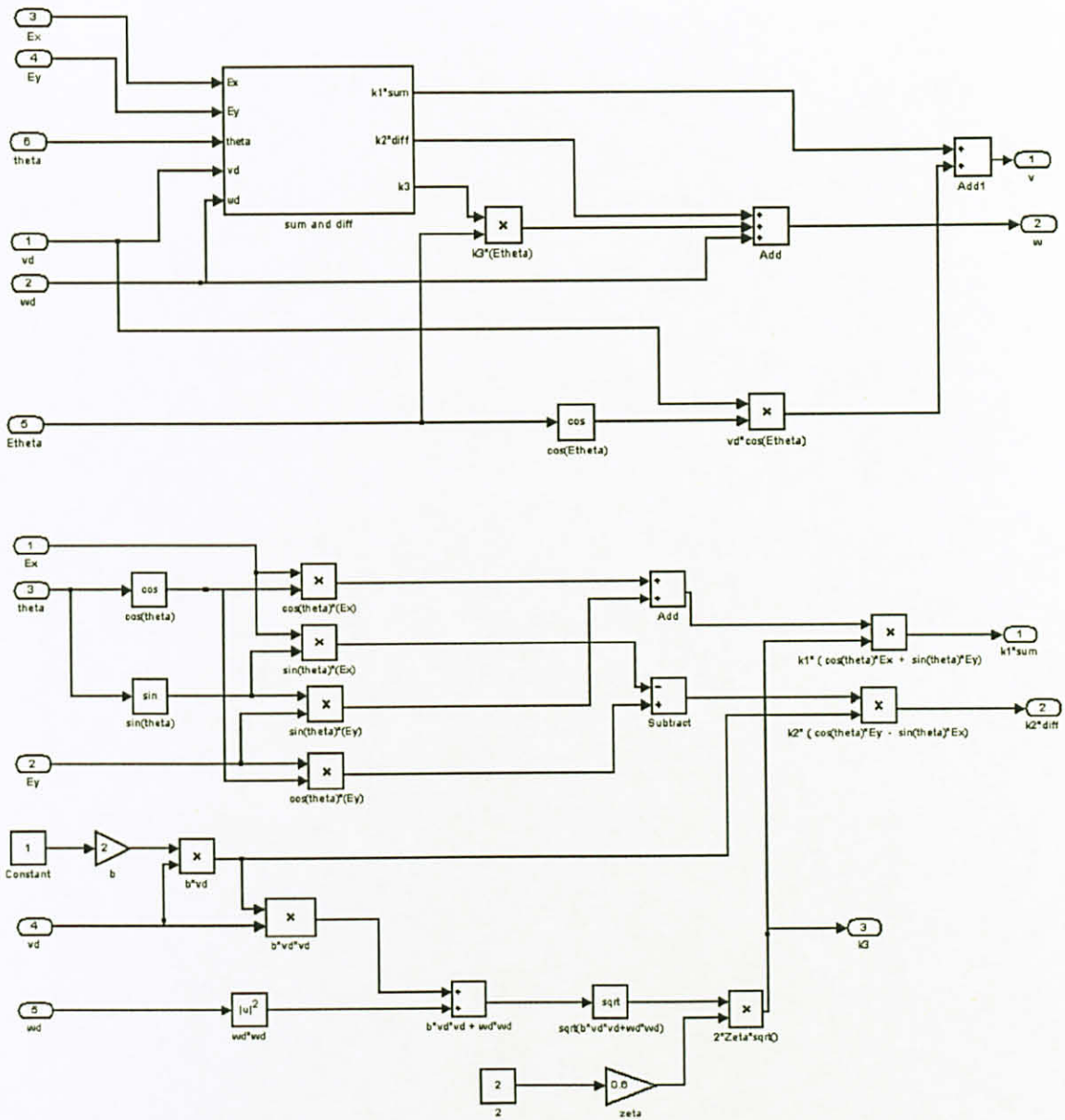


Figure 4.3: Feedback controller based on approximate linearization

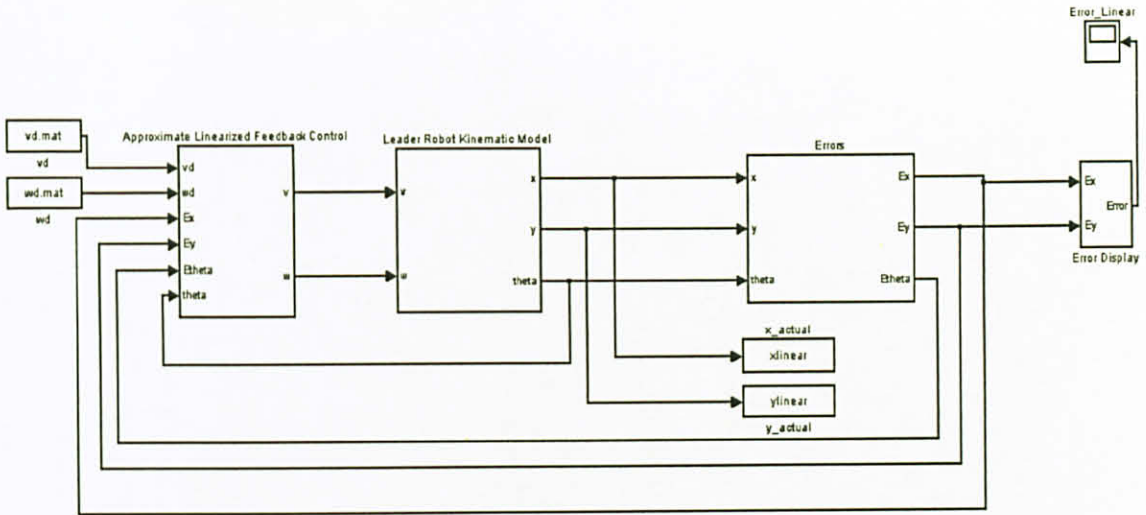


Figure 4.4: Leader robot using approximate linearization feedback controller

4.4 Feedback Controller using Cascaded Systems Theory

This controller was first proposed by [16]. The control law is given as

$$\begin{aligned} v &= v_d + c_2 e_1 - c_3 \omega_d e_2, & c_2 > 0, c_3 > -1 \\ \omega &= \omega_d + c_1 e_3, & c_1 > 0 \end{aligned} \quad (4.24)$$

The control law of Eq. 4.24 is K -exponentially stable if v_d is bounded and ω_d is persistently exciting. A small modification to this law was also proposed, which is

$$\begin{aligned} v &= v_d + c_2 e_1 - c_3 \omega_d e_2, & c_2 > 0, c_3 > -1 \\ \omega &= \omega_d + c_1 \sin e_3, & c_1 > 0 \end{aligned} \quad (4.25)$$

The control law of Eq. 4.25 results in local uniform exponential stable system if v_d is bounded and ω_d is persistently exciting. The feedback controller using cascaded systems theory is shown in Figure 4.5.

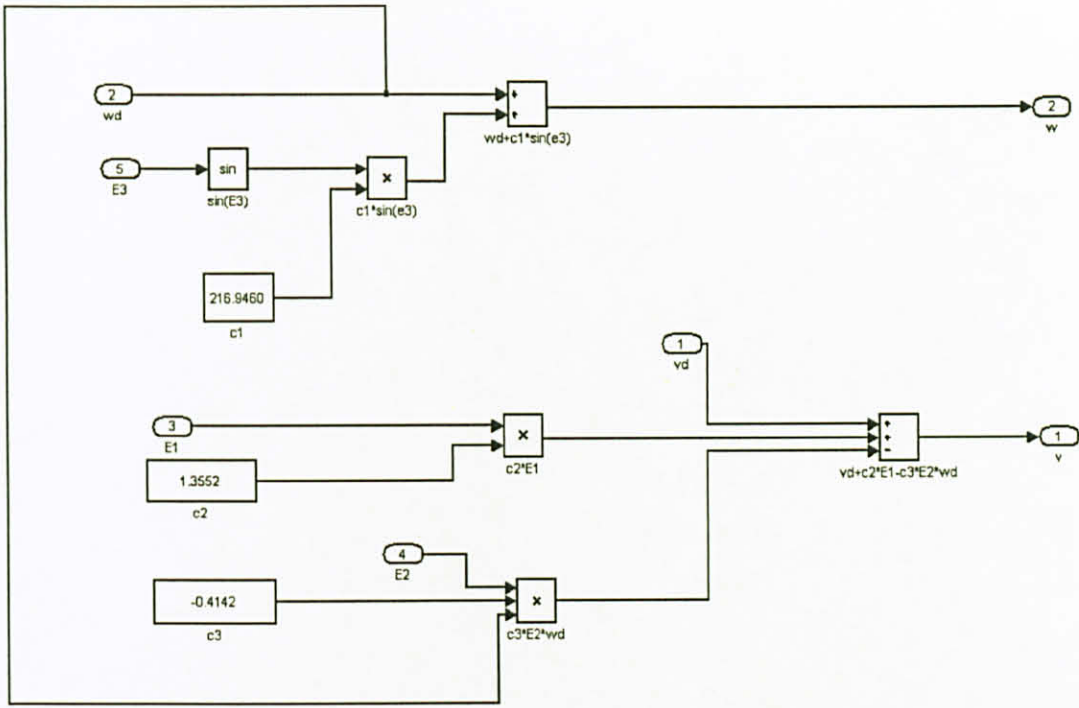


Figure 4.5: Feedback controller using cascaded systems theory

4.5 Feedback Controller using Linearization of Error Model

This controller was first proposed by [17]. The control law is given as

$$\begin{aligned} v &= v_d \cos e_3 + K_x e_1, & K_x &> 0 \\ \omega &= \omega_d + v_d (K_y e_2 + K_\theta \sin e_3), & K_y &> 0, K_\theta > 0 \end{aligned} \quad (4.26)$$

The stability analysis of the control law expressed in 4.26 states that if $v_d > 0$, then the system is locally asymptotically stable. Furthermore, if v_d and ω_d are both continuous, v_d , ω_d , K_x , K_θ are all bounded and if \dot{v}_d and $\dot{\omega}_d$ are both sufficiently small, then the system is locally uniformly asymptotically stable. The feedback controller using linearization of corresponding error model is shown in Figure 4.6.

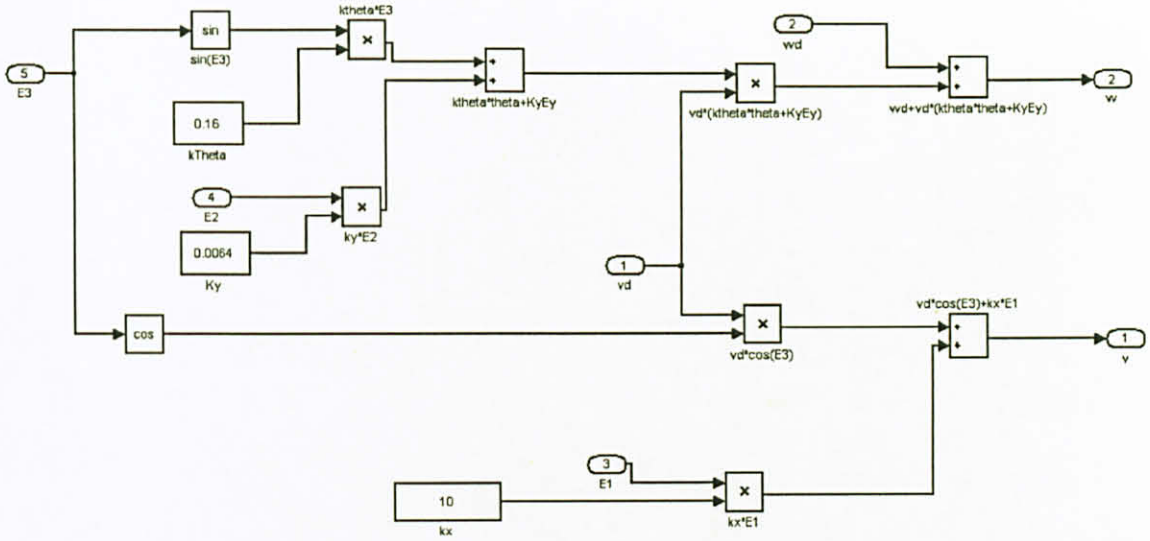


Figure 4.6: Feedback controller using linearization of error model

4.6 Feedback Controller using Nonlinear Design

The nonlinear design for feedback controller was first presented in [19]. The nonlinear feedback control strategy is expressed as follows.

$$\begin{aligned} u_1 &= -k_1(v_d(t), \omega_d(t))e_1 \\ u_2 &= -k_4 v_d(t) \frac{\sin e_3}{e_3} e_2 - k_3(v_d(t), \omega_d(t))e_3 \end{aligned} \quad (4.27)$$

where k_4 is a positive constant and k_1 and k_3 are continuous functions strictly positive in $\mathbb{R} \times \mathbb{R} - (0,0)$. The gains k_1 and k_3 are the same as Eq. 4.21, whereas k_4 is expressed as

$$k_4 = b, \quad b > 0 \quad (4.28)$$

Using the original control inputs, the control law is given by Eq. 4.29.

$$\begin{aligned} v &= v_d \cos(\theta_d - \theta) + k_1(v_d(t), \omega_d(t))[\cos \theta(x_d - x) + \sin \theta(y_d - y)] \\ \omega &= \omega_d + k_4 v_d \frac{\sin(\theta_d - \theta)}{\theta_d - \theta} [\cos \theta(y_d - y) - \sin \theta(x_d - x)] + \\ &\quad k_3(v_d, \omega_d)(\theta_d - \theta). \end{aligned} \quad (4.29)$$

The nonlinear control strategy globally asymptotically stabilizes the origin $e = 0$ which is demonstrated using Lyapunov stability theory. Assume the following Lyapunov function.

$$V = \frac{k_4}{2}(e_1^2 + e_2^2) + \frac{e_3^2}{2} \quad (4.30)$$

The time derivative of Eq. 4.30 is expressed as

$$\dot{V} = -k_1 k_4 e_1^2 + k_3 e_3^2 \leq 0 \quad (4.31)$$

Assuming $\|e(t)\|$ is bounded, $\dot{V}(t)$ is uniformly continuous and $V(t)$ tends to some limit value. Using Barbalat lemma, $\dot{V}(t)$ tends to zero. Thus if v_d and ω_d are bounded with bounded derivative, and that $v_d(t) \not\rightarrow 0$ or $\omega_d(t) \not\rightarrow 0$ when $t \rightarrow \infty$, the control law of Eq. 4.27 globally asymptotically stabilizes the origin $e = 0$ [19]. The feedback controller using nonlinear design is shown in Figure 4.7.

4.7 Full State Feedback Linearization via Static Feedback

The necessary condition for a system to be full state feedback linearized via static feedback is that the distribution, Δ , generated by system vector fields must be involutive [30]. Applying this condition to the leader robot modeled by Eq. 4.15, the full state feedback linearization cannot be achieved using static (time invariant) feedback. The reason for this is that Δ for the system is not involutive as explained in Section 2.4.

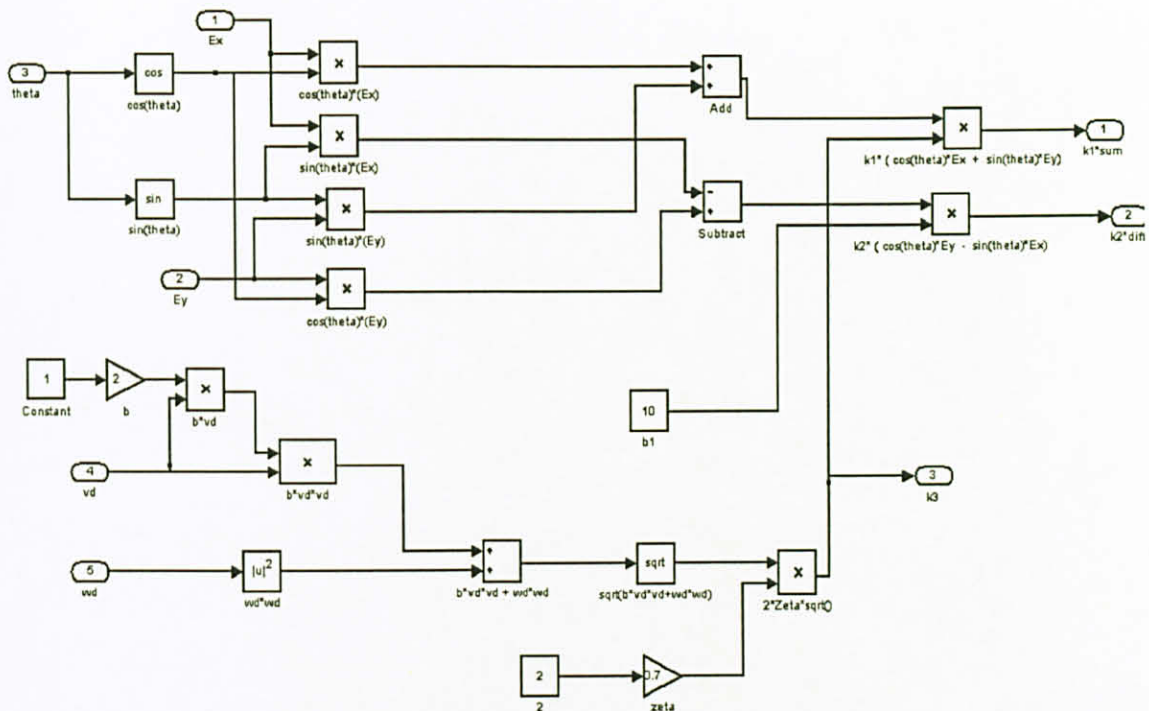


Figure 4.7: Feedback controller using nonlinear design

4.8 Input-output Feedback Linearization via Static Feedback

If the static feedback design for full state feedback fails, input-output linearization may be possible. The main idea of input-output linearization is to transform m equations via feedback into simple decoupled integrators, where m represents the number of inputs. However the choice of outputs which are linearized is not unique. For the leader robot, the outputs are chosen as $z = [z_1, z_2]^T$, which can be expressed as

$$z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} \quad (4.32)$$

The linearization algorithm begins by computing the derivative of the output as

$$\dot{z} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} = A(\theta)u \quad (4.33)$$

where $A(\theta)$ is the decoupling matrix. Since $A(\theta)$ is singular, static feedback fails to achieve input-output linearization. However a possible way is to redefine the system outputs at point D , as shown in Figure 4.1. The system output is expressed as

$$z = \begin{bmatrix} x + d \cos \theta \\ y + d \sin \theta \end{bmatrix} \quad (4.34)$$

Differentiating Eq. 4.34, the following is obtained.

$$\dot{z} = \begin{bmatrix} \cos \theta & -b \sin \theta \\ \sin \theta & b \cos \theta \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} = A(\theta)u \quad (4.35)$$

Since $\det(A(\theta)) = b \neq 0$, let $\dot{z} = r$, where r is an auxiliary input. Solving for the inputs, the following is obtained

$$u = A^{-1}(\theta)r \quad (4.36)$$

In terms of the transformed coordinates (z_1, z_2, θ) , the close-loop system becomes

$$\begin{aligned} \dot{z}_1 &= r_1 \\ \dot{z}_2 &= r_2 \\ \dot{\theta} &= \frac{r_2 \cos \theta - r_1 \sin \theta}{d} \end{aligned} \quad (4.37)$$

which is input-output linearized and decoupled.

The control law for the inputs is given as

$$r_i = z_{di} + k_{pi}(z_{di} - z_i), \quad k_{pi} > 0, \quad i = 1, 2 \quad (4.38)$$

The control law of Eq. 4.38 achieves exponential convergence of the output tracking error to zero [7], [14], [30]. However the following conclusions can be made.

- If the system outputs are defined as in Eq. 4.34, there are two options for generating the reference input trajectory. The first option is to directly plan a cartesian motion to be executed by point D . The second option is that the trajectory planner generates a desired motion for flat outputs $[x_d(t), y_d(t)]^T$, associated with the inputs, $[v_i, \omega_i]^T$. If the second case is considered, the trajectory needs to be converted into a reference motion for the point D .
- A complete analysis would require the study of the stability of the time-varying system, modeled by Eq. 4.37, with the input r given by Eq. 4.38.
- The output choice, Eq. 4.34, is not the only one leading to input output linearization and decoupling static feedback. Another possible choice for the output variables can be the chained form transformations of Eq. 3.32.

4.9 Full State Feedback Linearization via Dynamic Feedback

In order to design a trajectory tracking controller directly for the cartesian coordinates (x, y) , dynamic extension is required to overcome the singularity of the decoupling matrix of Eq. 4.32. The dynamic state feedback compensator is given as

$$\begin{aligned} v &= a(q, \xi) + b(q, \xi)r \\ \xi &= c(q, \xi) + d(q, \xi)r \end{aligned} \quad (4.39)$$

where $\xi(t) \in \mathbb{R}^v$ is the compensator state vector of dimensions v , and $r(t) \in \mathbb{R}^v$ is the auxiliary input. The condition for choosing $\xi(t)$ and $r(t)$ are such that the close-loop system of Eq. 2.4 and Eq. 4.39 are equivalent, under a state transformation $z = T(q, \xi)$, to a linear controllable system. For the nonholonomic leader robot, the linearization process involves the following procedure.

Initially, the system output is defined. To this output, a desired behavior is assigned such as track a trajectory. This output is successively differentiated until the system inputs appear explicitly in a nonsingular way. If in a step involving differentiation of system outputs, the decoupling matrix of the system is singular (which means that some of the inputs are still not appearing), integrators are added on some of the inputs and the process of differentiation is continued. This operation is known as dynamic extension. The dynamic compensator has the new auxiliary input, r , as its input. The process of differentiation is continued until at some point, the system is invertible. The number of successive addition of integrators gives dimension of the state ξ of the dynamic compensator. For the system modeled by Eq. 4.15, the output is defined as

$$z = \begin{bmatrix} x \\ y \end{bmatrix}, \quad \dot{z} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (4.40)$$

From Eq. 4.40, it can be observed that only v affects \dot{z} , while ω cannot be recovered. In order to proceed, an integrator, ξ , is added on the linear velocity input v , as

$$\xi = v, \quad \dot{\xi} = a \quad (4.41)$$

where a is the new input representing the linear acceleration of the leader robot. In terms of ξ , Eq. 4.40 can be expressed as

$$\dot{z} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \end{bmatrix} \begin{bmatrix} \xi \\ \omega \end{bmatrix} \Rightarrow \dot{z} = \xi \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} \quad (4.42)$$

Differentiating Eq. 4.41, the following is obtained

$$\ddot{z} = \dot{\xi} \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} + \xi \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix} \dot{\theta} \quad (4.43)$$

Substituting the value of $\dot{\xi}$ from Eq. 4.41 and $\dot{\theta} = \omega$, the following is obtained

$$\ddot{z} = a \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} + \xi \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix} \omega = \begin{bmatrix} \cos \theta & -\xi \sin \theta \\ \sin \theta & \xi \cos \theta \end{bmatrix} \begin{bmatrix} a \\ \omega \end{bmatrix} \quad (4.44)$$

From Eq. 4.44, it can be observed that the decoupling matrix multiplied with the modified input (a, ω) is nonsingular provided that $\xi \neq 0$.

Let $\ddot{z} = r$, so the inputs can be obtained as

$$\begin{bmatrix} a \\ \omega \end{bmatrix} = \begin{bmatrix} \cos \theta & -\xi \sin \theta \\ \sin \theta & \xi \cos \theta \end{bmatrix}^{-1} \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta / \xi & \cos \theta / \xi \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} \quad (4.45)$$

Substituting the values for original inputs, the resulting dynamic compensator and the inputs are

$$\begin{aligned} v &= \xi \\ \omega &= \frac{r_2 \cos \theta - r_1 \sin \theta}{\xi} \\ \dot{\xi} &= r_1 \cos \theta + r_2 \sin \theta \end{aligned} \quad (4.46)$$

As one integrator, ξ , was added, hence the order of the dynamic compensator is one.

The new coordinates can be written as

$$\begin{aligned} z_1 &= x \\ z_2 &= y \\ z_3 &= \dot{x} = \xi \cos \theta \\ z_4 &= \dot{y} = \xi \sin \theta \end{aligned} \tag{4.47}$$

The extended system of Eq. 4.47 is fully linearized in a controllable form. The decoupled chain of input output integrators can be written as

$$\begin{aligned} \ddot{z}_1 &= r_1 \\ \ddot{z}_2 &= r_2 \end{aligned} \tag{4.48}$$

The dynamic compensator of Eq. 4.46 has a potential singularity when $\xi = v = 0$. Assuming that the robot must follow a smooth output trajectory $[x_d(t), y_d(t)]^T$, which is persistent and the linear velocity, v , does not go to zero. The globally exponentially stabilizing feedback law for the trajectory is given as

$$\begin{aligned} r_1 &= \ddot{x}_d(t) + k_{p1}(x_d(t) - x) + k_{d1}(\dot{x}_d(t) - \dot{x}) \\ r_2 &= \ddot{y}_d(t) + k_{p2}(y_d(t) - y) + k_{d2}(\dot{y}_d(t) - \dot{y}) \end{aligned} \tag{4.49}$$

with PD gains chosen as $k_{pi} > 0$, $k_{di} > 0$, for $i = 1, 2$. The full state feedback linearized controller is shown in Figure 4.8. The values of \dot{x} and \dot{y} can be computed from Eq. 4.47 as a function of the robot state and the compensator state, ξ . The values of the feedback gains are chosen such that the polynomial expressed by Eq. 4.50 is Hurwitz.

$$\lambda^2 + k_{di}\lambda + k_{pi}, \quad i = 1, 2 \tag{4.50}$$

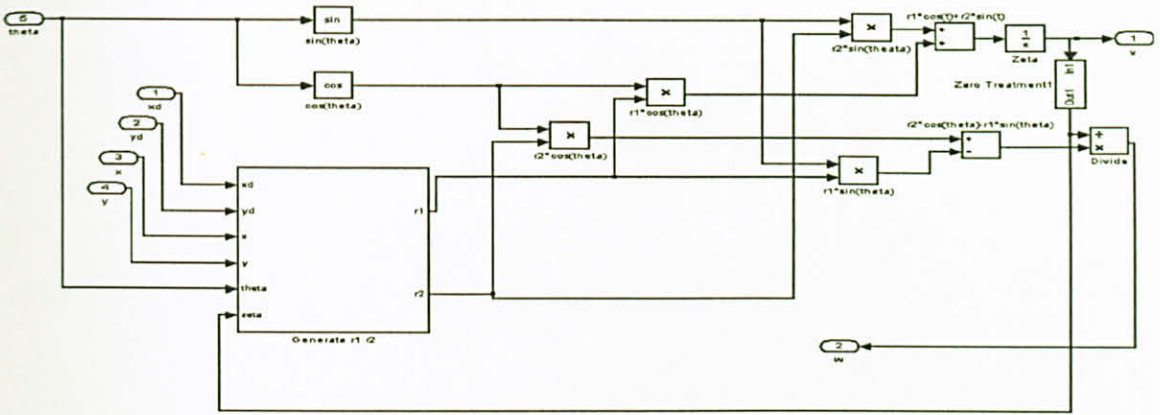


Figure 4.8: Full state linearized via dynamic feedback controller

4.10 Posture Stabilization

In this section, the point to point motion for the collaborative robots is discussed. The objective is to reach a final desired configuration starting from an initial point, without the need to plan a trajectory. As stated in Section 3.1, the collaborative robot system is not point stabilizable at a point by smooth continuous feedback. Therefore, the available techniques are to use smooth time-varying feedback, non smooth time-varying feedback and design based on polar coordinates.

4.10.1 Smooth Time-varying Feedback

Using the nonlinear feedback control design in Section 4.6, asymptotic stabilization of the state tracking error can be achieved provided that $v_d(t)$ and $\omega_d(t)$ do not both vanish to zero in finite time. The smooth time-varying feedback controller for posture stabilization is similar to the nonlinear feedback controller and is expressed as

$$\begin{aligned} u_1 &= -k_1(v_d(t), \omega_d(t))e_1, \\ u_2 &= -k_2 v_d(t) \frac{\sin e_3}{e_3} e_2 - k_3(v_d(t), \omega_d(t))e_3 \end{aligned} \quad (4.51)$$

where k_1 , k_2 and k_3 are positive constants. For generating the control inputs, Eq. 4.29 can be used. The control law of Eq. 4.51 globally asymptotically stabilizes the origin $e = 0$ as discussed in section 4.6.

4.10.2 Design based on Polar Coordinates

This control design is based on the change of coordinates. Let ψ be the distance of the reference point (x, y) of the leader robot from the goal, μ be the angle of the pointing vector to the goal with respect to the robot main axis and ϕ be the angle of the same pointing vector with respect to the x -axis of the robot. The leader robot expressed in the polar coordinates is shown in Figure 4.9.

The state transformation is given as

$$\begin{aligned}\psi &= \sqrt{x^2 + y^2} \\ \mu &= \tan^{-1}(y/x) - \theta + \pi \\ \phi &= \mu + \theta\end{aligned}\tag{4.52}$$

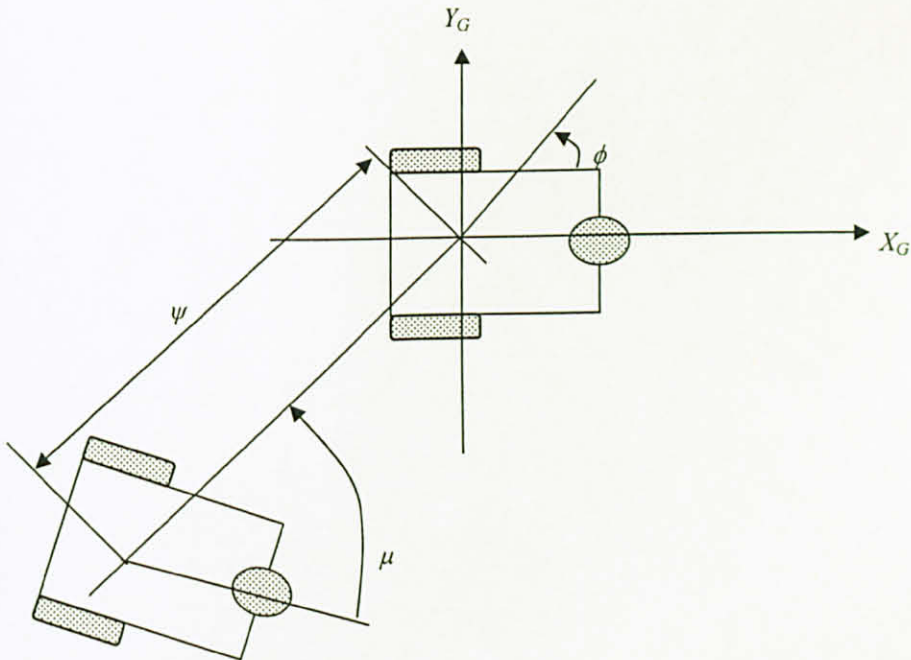


Figure 4.9: Leader robot positioning in polar coordinates

Differentiating Eq. 4.52, the transformed kinematic equations can be written as

$$\begin{aligned}\dot{\psi} &= -v \cos \mu \\ \dot{\mu} &= \frac{\sin \mu}{\psi} v - \omega \\ \dot{\phi} &= v \frac{\sin \mu}{\psi}\end{aligned}\quad (4.53)$$

The control law for posture stabilization based on the polar coordinates was proposed in [24]. The control law is given by Eq. 4.54

$$\begin{aligned}v &= k_1 \psi \cos \mu \\ \omega &= k_2 \mu + k_1 \frac{\sin \mu \cos \mu}{\mu} (\mu + k_3 \phi)\end{aligned}\quad (4.54)$$

The feedback law of Eq 4.54 globally asymptotically stabilizes the origin [24], [31].

The posture stabilization controller of Eq. 4.54 is shown in Figure 4.10.

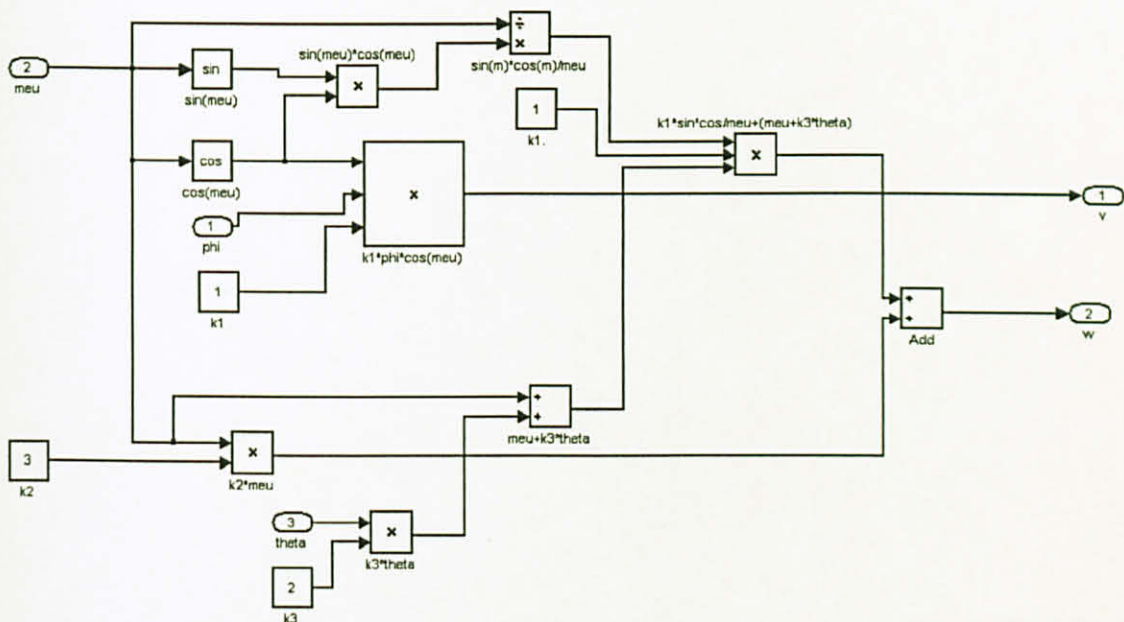


Figure 4.10: Posture stabilization controller based on polar coordinates

4.10.2 Dynamic Feedback Linearized Controller

The dynamic feedback linearized controller can be extended to address the issue of posture stabilization. The feedback control law is given as in [25]

$$\begin{aligned} r_1 &= -k_{p1}(x) - k_{d1}(\dot{x}) \\ r_2 &= -k_{p2}(y) - k_{d2}(\dot{y}) \end{aligned} \quad (4.55)$$

where k_{p1} , k_{p2} , k_{d1} , k_{d2} are the feedback gains. The feedback law of Eq. 4.55 yields exponential convergence from any initial configuration to the origin [25]. The dynamic feedback controller for posture stabilization is shown in Figure 4.11.

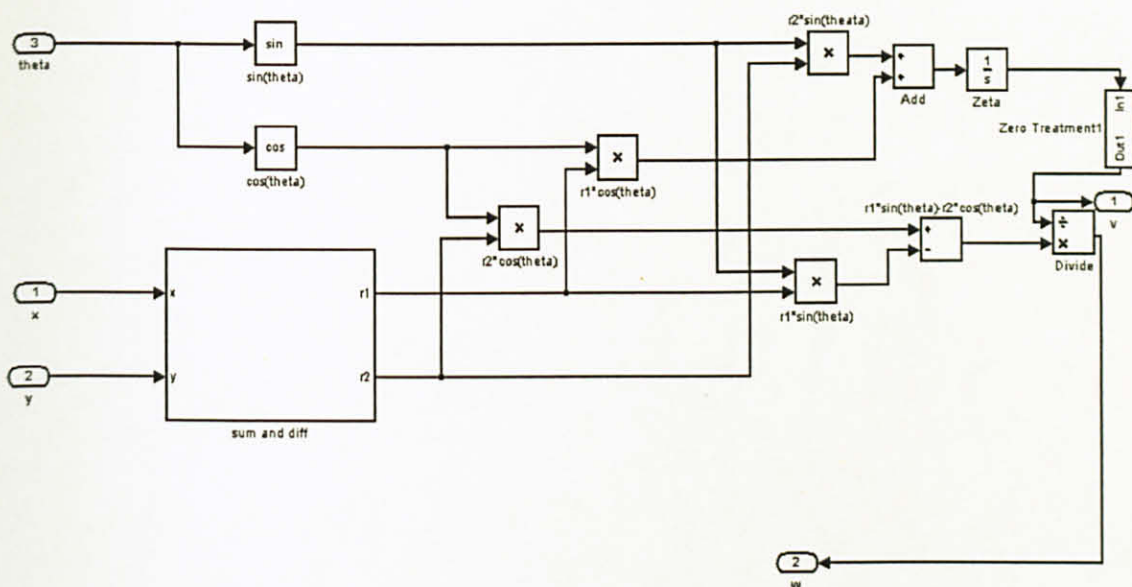


Figure 4.11: Posture stabilization controller based on dynamic feedback

4.11 Feedback Linearized Control Strategies for Follower Robots

In this section, feedback linearized control strategies for the follower robots are presented. The feedback laws are presented for different formation controllers as follows.

4.11.1 Feedback Strategy for Separation Bearing Controller

The kinematic model for the follower robot using the separation bearing controller was expressed in Eq. 2.25 and Eq. 2.26. To recall, the kinematic model in compact form is given as

$$\begin{aligned}\dot{z}_{lf} &= G_{SB}(z_{lf}, \gamma)u_f + F_{SB}(z_{lf})u_l \\ \dot{\theta}_f &= \omega_f\end{aligned}\tag{4.56}$$

where $z_{lf} = (l_{lf}, \varphi_{lf})$, $u_f = (v_f, \omega_f)$, $u_l = (v_l, \omega_l)$ and

$$\begin{aligned}G_{SB} &= \begin{bmatrix} \cos \gamma & d \sin \gamma \\ -\sin \gamma & d \cos \gamma \\ l_{lf} & l_{lf} \end{bmatrix} \\ F_{SB} &= \begin{bmatrix} -\cos \varphi_{lf} & 0 \\ \sin \varphi_{lf} / l_{lf} & -1 \end{bmatrix}\end{aligned}\tag{4.57}$$

The input-output linearization technique begins by defining the output as $z_{lf} = (l_{lf}, \varphi_{lf})$. Differentiating the output, Eq. 4.58 is obtained.

$$\dot{z}_{lf} = G_{SB}(z_{lf}, \gamma)u_f + F_{SB}(z_{lf})u_l = A(z_{lf})u_f + B\tag{4.58}$$

The determinant of the decoupling matrix, $A(z_{lf})$, is $d/l \neq 0$.

Since $A(z_{lf})$ is nonsingular, the control velocities for the follower robot can be expressed as

$$u_f = G_{SB}^{-1}(p_{SB} - F_{SB}u_l) \quad (4.59)$$

where p_{SB} is an auxiliary control input given as

$$p_{SB} = K \tilde{z}_{lf} = \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} \begin{bmatrix} l_{lf}^d - l_{lf} \\ \varphi_{lf}^d - \varphi_{lf} \end{bmatrix} \quad (4.60)$$

with $k_1, k_2 > 0$ as the controller gains. The control inputs for the follower robot are expressed by Eq. 4.61.

$$\begin{aligned} v_f &= \rho - d\omega_f \tan \gamma \\ \omega_f &= \frac{\cos \gamma}{d} \{k_a l_{lf} (\varphi_{lf}^d - \varphi_{lf}) - v_l \sin \varphi_{lf} + l_{lf} \omega_l + \rho \sin \gamma\} \\ \text{where} \quad & \quad \quad \quad (4.61) \\ \rho &= \frac{k_b (l_{lf}^d - l_{lf}) + v_l \cos \varphi_{lf}}{\cos \gamma} \\ \gamma &= \varphi_{lf} + \theta_l - \theta_f \end{aligned}$$

The stability of the controller expressed by Eq. 4.61 was presented in [21], [22]. If the linear velocity of the leader robot is lower bounded *i.e.* $v_l > 0$, angular velocity is bounded *i.e.* $\omega_l < \omega_{\max}$, and the initial orientation is such that $|\theta_l(0) - \theta_f(0)| < \pi$, then the system of Eq. 4.61 is stable and the output error converges to zero exponentially. The input-output linearized feedback strategy for the follower robot using separation bearing controller is shown in Figure 4.12.

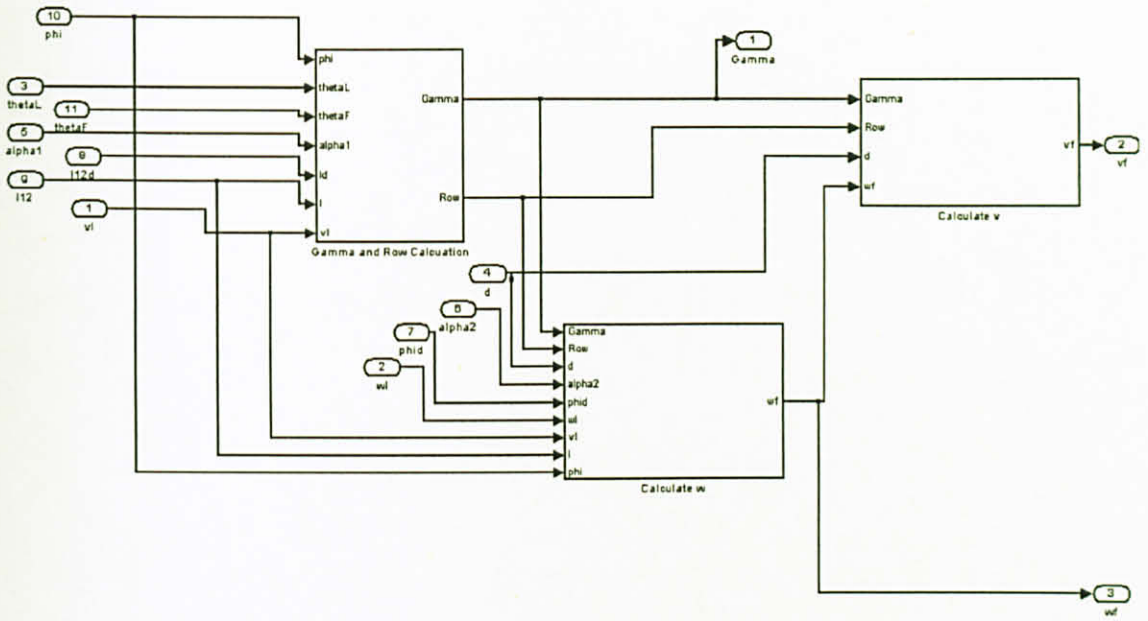


Figure 4.12: Input-output linearized feedback strategy for follower robot using separation bearing formation

4.11.2 Feedback Strategy for Separation-separation Controller

The kinematic model for the follower robot using the separation-separation controller was expressed using Eq. 2.30. Using input-output linearization techniques the control law for the follower robot is given as

$$\begin{aligned}
 v_f &= \frac{k_1(l_{1f}^d - l_{1f}) + v_1 \cos \varphi_{1f} - d\omega_f \sin \gamma_1}{\cos \gamma_1} \\
 \omega_f &= \frac{k_1(l_{1f}^d - l_{1f}) \cos \gamma_2 + v_1 \cos \varphi_{1f} \cos \gamma_2}{d \sin(\gamma_1 - \gamma_2)} + \\
 &\quad \frac{v_2 \cos \varphi_{2f} \cos \gamma_1 - k_2(l_{2f}^d - l_{2f}) \cos \gamma_1}{d \sin(\gamma_1 - \gamma_2)}
 \end{aligned} \tag{4.62}$$

where

$$\gamma_i = \varphi_{if} + \theta_i - \theta_f, \quad i = 1, 2$$

The stability analysis of the controller expressed in Eq. 4.62 was presented in [21], [22]. Assuming the linear velocity of the leader robot 1 is lower bounded *i.e.* $v_1 > 0$, angular velocity is bounded *i.e.* $\omega_1 < \omega_{\max}$, and the initial relative orientation is such that $|\theta_1(0) - \theta_i(0)| < \pi$ with $i = 2, f$. If the control input u_{2f} is obtained using feedback linearization, then the system of Eq. 4.62 is stable and the output converges exponentially to the desired value z_d . The input-output linearized feedback strategy for the follower robot using separation-separation controller is shown in Figure 4.13.

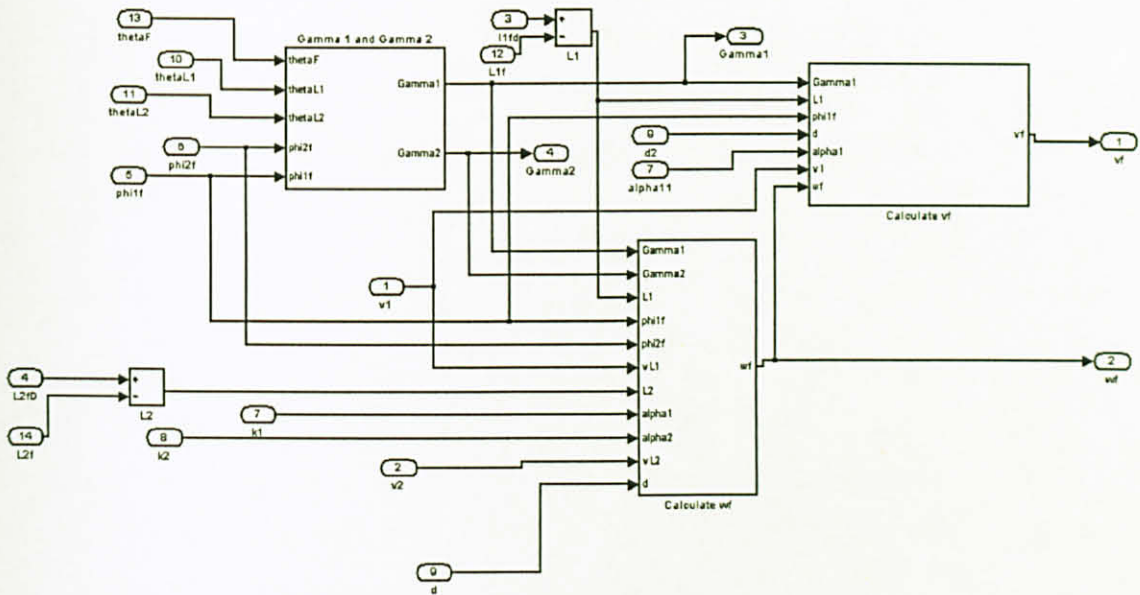


Figure 4.13 : Input-output linearized feedback strategy for follower robot using separation-separation formation

4.12 Summary

In this chapter, feedback controllers for the leader and follower robots were presented. The full state and input-output feedback linearization techniques were applied to the leader and follower models. The stability analysis of the feedback controllers was also discussed. The posture stabilization controllers for the leader robot were also presented. In the next chapter, these feedback controllers are analyze and simulated for a different set of trajectories.

CHAPTER 5

SIMULATION RESULTS

This chapter presents the simulation results for the different feedback control strategies applied to the leader-follower formation. The feedforward input command controller for the leader robot is derived. A framework for the collaborative robots is presented and modeled using MATLAB/Simulink. The framework consists of feedforward and feedback controllers. The feedback linearized and the posture stabilization controllers are simulated for a given set of trajectories. From the simulation results, it is observed that the dynamic feedback linearized control strategy for the leader robot, and input-output linearized feedback strategy for the follower robots minimize the error more efficiently than other strategies for the given trajectories.

5.1 Feedforward Command Controller

Assuming that the leader robot follows a desired cartesian trajectory $[x_d(t), y_d(t)]^T$ with $t \in [0, T]$. As stated in section 3.5, using the flat outputs for the system, the remaining states as well as the control inputs can be computed by algebraic transformations. The flat outputs for the leader robot model are $[x(t), y(t)]^T$. Knowing the desired flat outputs, θ_d can be calculated as

$$\theta_d = \text{atan2}(\dot{y}_d, \dot{x}_d) \quad (5.1)$$

where atan2 is the fourth-quadrant inverse tangent and is undefined only if both arguments are zero. Differentiating Eq. 2.15 with respect to time, the feedforward control inputs are computed as

$$v_d(t) = \pm \sqrt{\dot{x}_d^2(t) + \dot{y}_d^2(t)} \quad (5.2)$$

$$w_d(t) = \frac{\ddot{y}_d(t)\dot{x}_d(t) - \ddot{x}_d(t)\dot{y}_d(t)}{\dot{x}_d^2(t) + \dot{y}_d^2(t)} \quad (5.3)$$

The sign for $v_d(t)$ will determine the forward or backward motion of the robots. Eq. 5.3 is not defined when $v_d(t)$ is equal to zero. The feedforward command controller is modeled using Simulink as is shown in Figure 5.1.

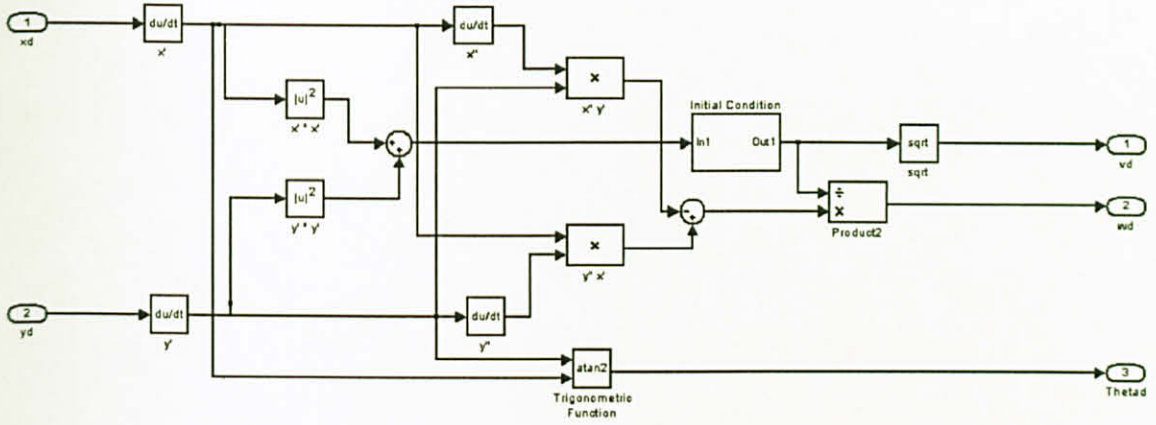


Figure 5.1: Feedforward command controller for the leader robot

5.2 Framework for Collaborative Robots

The simulation testbed is implemented using MATLAB/Simulink. The Bluetooth USB dongles are configured to form the piconet. Each dongle is connected to a computer. A MATLAB/Simulink session runs on each computer. Each MATLAB session models the leader-follower formation for the leader and follower robots. The master robot in the piconet acts as the leader and the slaves act as follower robots. The framework for the leader and follower robots is shown in Figure 5.2 and 5.3, respectively.

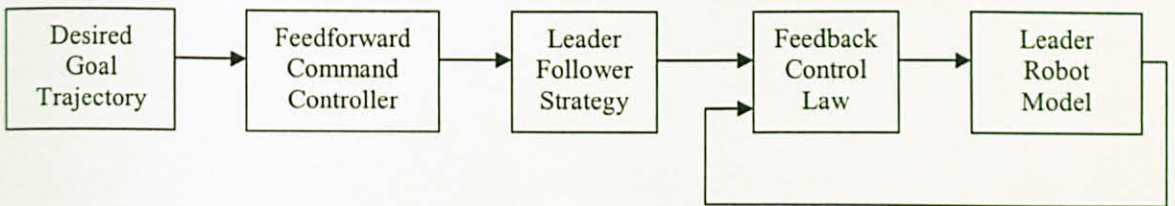


Figure 5.2: Framework for the leader robot

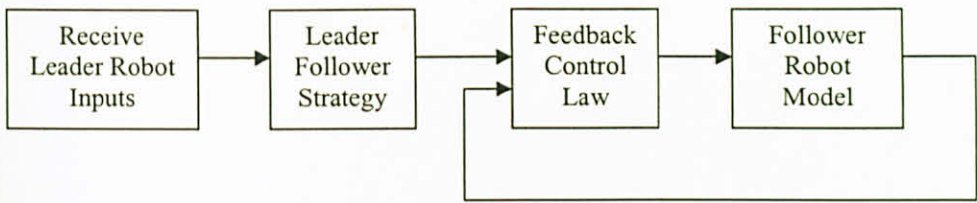


Figure 5.3: Framework for the follower robot

For a desired and feasible goal trajectory, $[x_d(t), y_d(t)]^T$, the feedforward command controller generates the feedforward control inputs, $[v_d, \omega_d]^T$, for the leader robot. Using the leader-follower strategy, the leader robot transmits the control inputs to the follower robots. The control inputs are transmitted using the Bluetooth piconet. The follower robots receive the leader robot inputs and derive their own control inputs, $[v_f, \omega_f]^T$, using the leader-follower formation control. The control inputs for both the leader and follower robots are fed into the feedback control law. The feedback control strategy generates the actual inputs based on the feedforward inputs and feedback states of the robots.

A feasible trajectory must satisfy the nonholonomic constraint for the collaborative robots. It means that the state θ and the inputs can be recovered from the trajectory. A feasible trajectory must be twice differentiable in order to generate the control inputs. Figure 5.4 show examples of non-feasible trajectory for the nonholonomic robots.

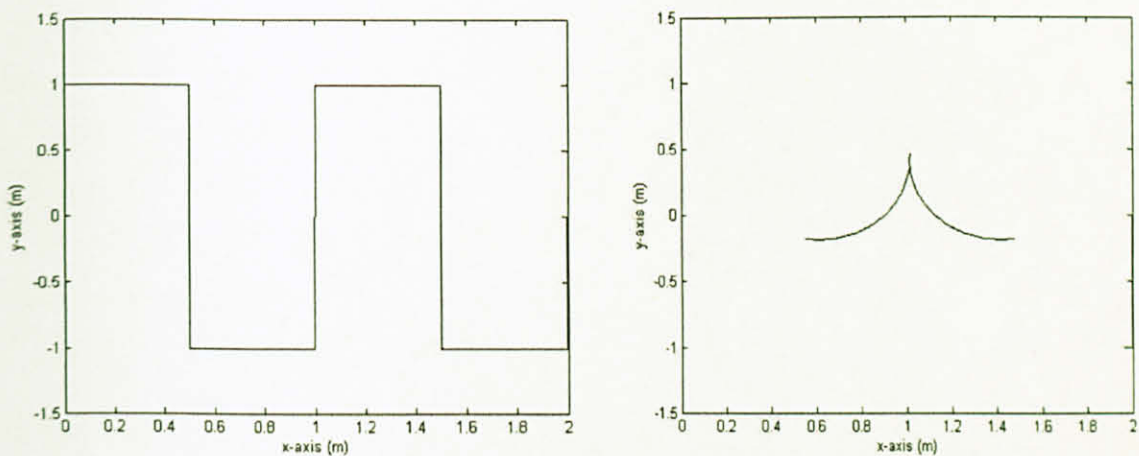


Figure 5.4: Non-feasible trajectories for nonholonomic robots

5.3 Simulation Results for the Leader Robot

Using the leader-follower formation for collaborative robots, the following different trajectories were considered for the leader robot.

5.3.1 Test 1 (Eight Shaped Trajectory)

In the first test, the desired trajectory was defined as follows.

$$x_d(t) = 10 \sin(t/20), y_d(t) = 10 \sin(t/40) \quad (5.4)$$

The desired trajectory of Eq. 5.4 begins at the origin (0, 0) and completes a full cycle when $T = 2\pi(40) \approx 251.3$ sec. This trajectory is shown in Figure 5.5. The linear and angular velocities inputs for the leader robot are shown in Figure 5.6 and 5.7, respectively. The error norm for the actual trajectory using approximate linearized controller, nonlinear controller, cascaded systems controller, stable tracking controller and full state dynamic feedback linearized controller are shown in Figure 5.8, 5.9, 5.10, 5.11 and 5.12, respectively. The actual trajectory using dynamic feedback and nonlinear controller is shown in Figure 5.13. The values for different parameters in the feedback controllers are listed in Table 5.1. The error statistics for the given trajectory are summarized in Table 5.2.

Table 5.1: Parameters values for different feedback controllers

Controllers	Parameters values
Feedforward	$v_d(0) = 0.0125 \text{ m/sec}$
Approximate linearized feedback	$\zeta = 0.5, b = 2$
Nonlinear feedback	$\zeta = 0.5, b = 2$
Cascaded systems feedback	$c_1 = 216.9, c_2 = 1.355$ and $c_3 = -0.414$
Stable tracking	$K_x = 10, K_y = 0.0064$ and $K_\theta = 0.16$
Dynamic linearized feedback	$k_{d1} = k_{d2} = 0.7, k_{p1} = k_{p2} = 1, \zeta(0) = v(0)$

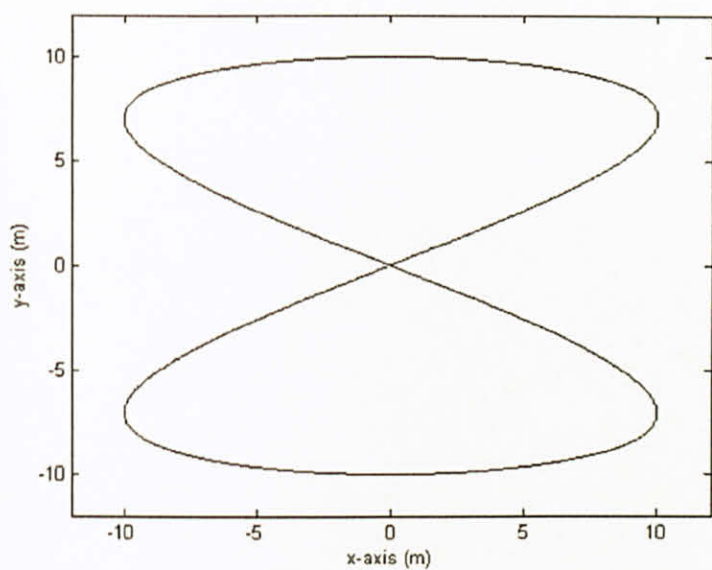


Figure 5.5: Desired trajectory for test 1

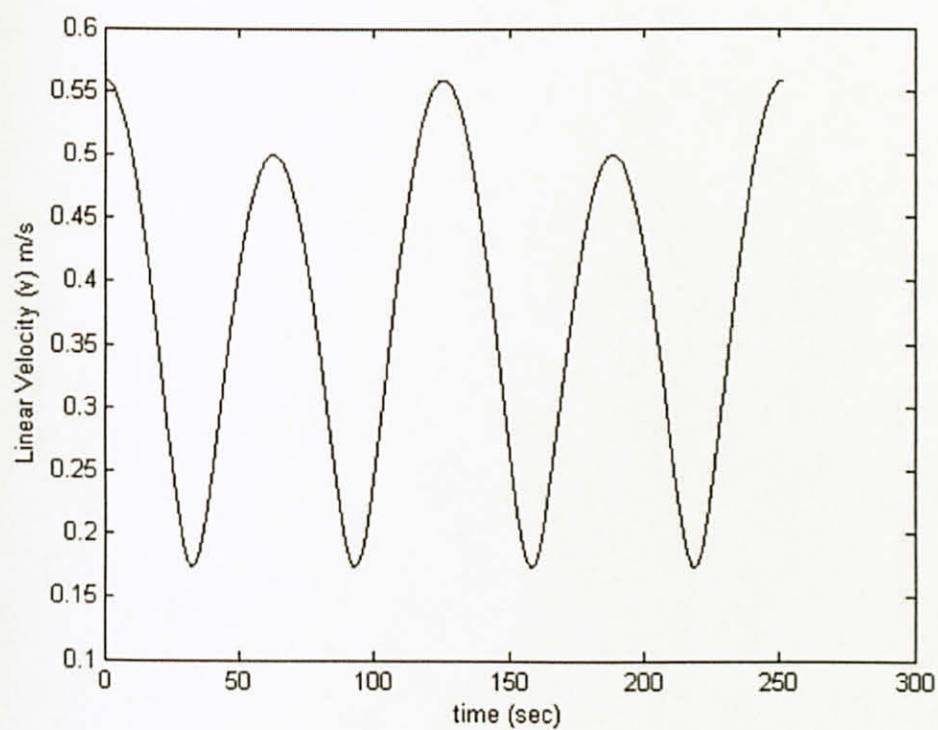


Figure 5.6: Linear velocity for the desired trajectory of test 1

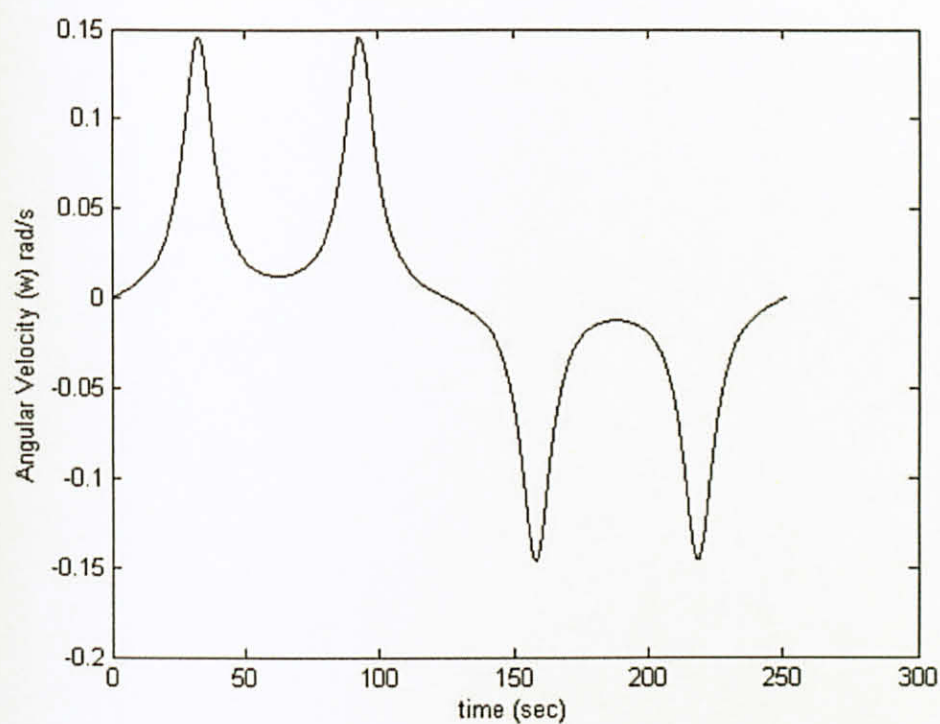


Figure 5.7: Angular velocity for the desired trajectory of test 1

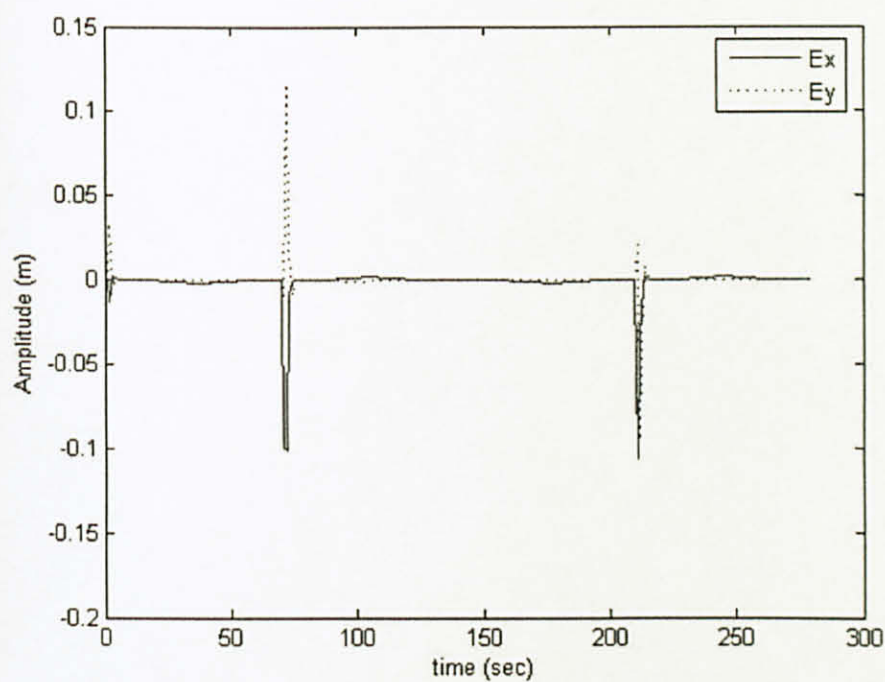


Figure 5.8: Norm of error for the trajectory of test 1 using approximate linearized controller

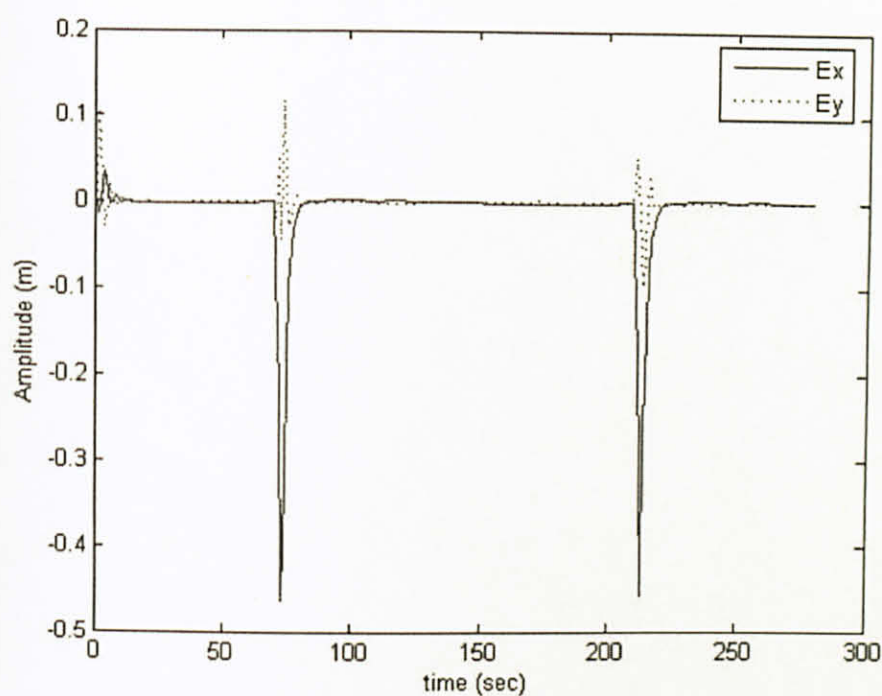


Figure 5.9: Norm of error for the trajectory of test 1 using nonlinear controller

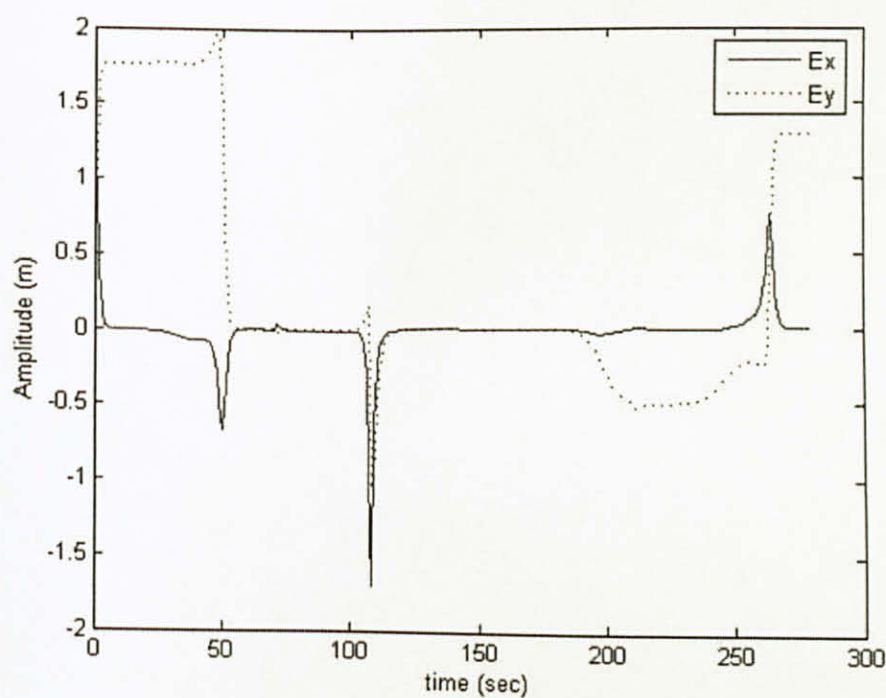


Figure 5.10: Norm of error for the trajectory of test 1 using cascaded systems controller

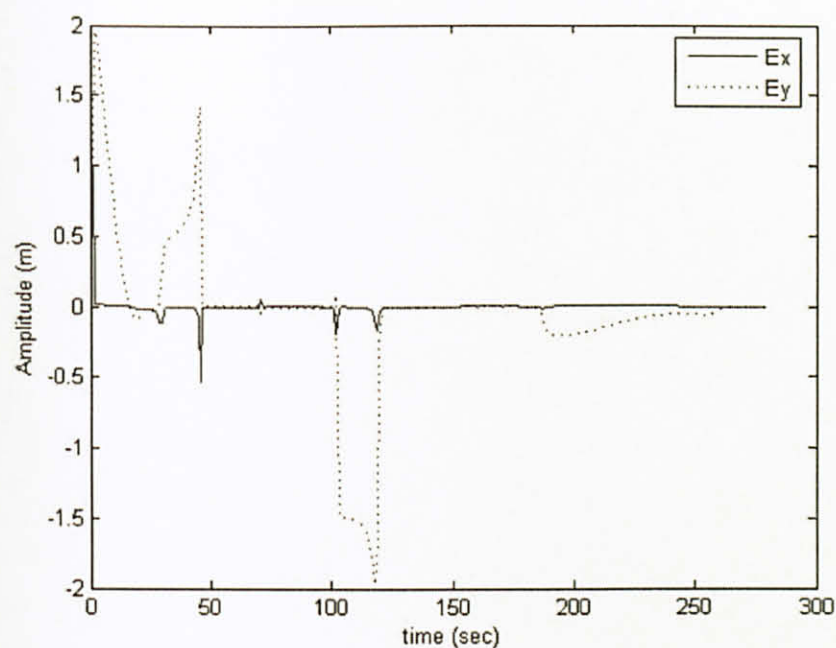


Figure 5.11: Norm of error for the trajectory of test 1 using stable tracking controller

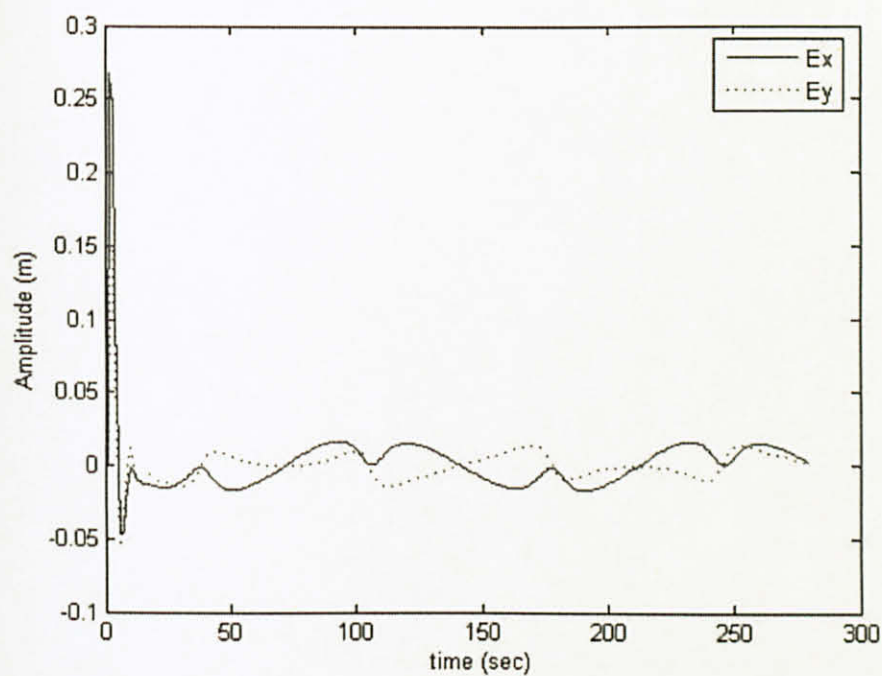


Figure 5.12: Norm of error for the trajectory of test 1 using full state linearized via dynamic feedback controller

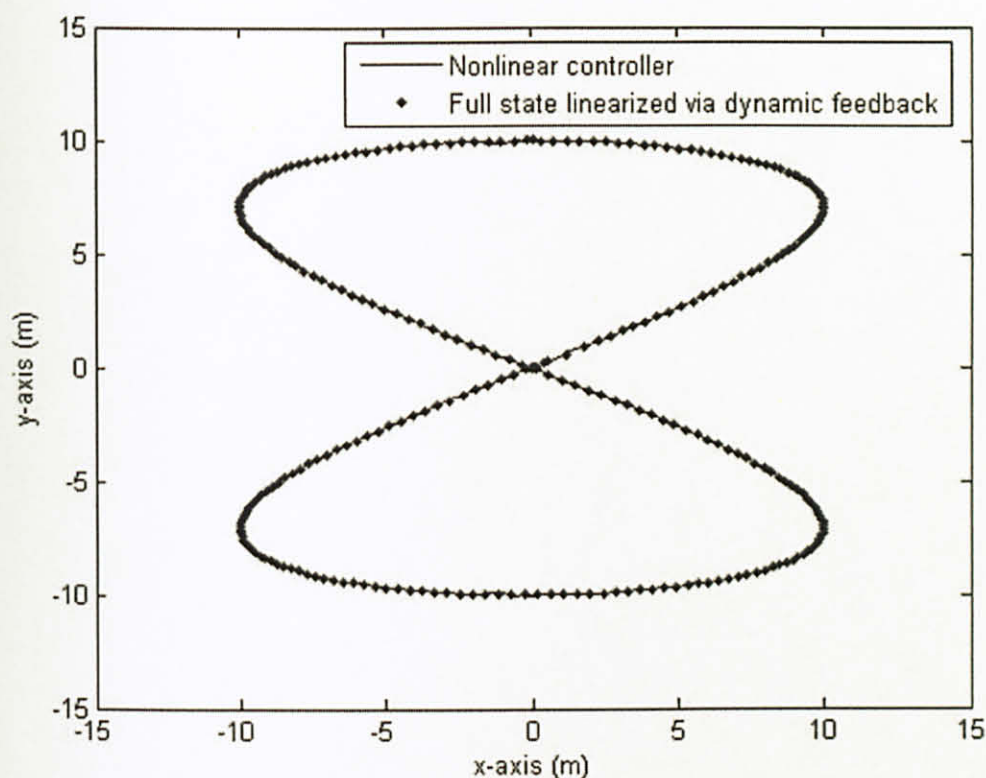


Figure 5.13: Actual trajectory by the leader robot for test 1 using full state linearized via dynamic feedback and nonlinear controller

Table 5.2: Error statistics using different feedback controllers for test 1

Statistical parameter	Mean (m)	Standard Deviation (m)	Variance (m)
Approximate linearized	0.1622	0.7746	0.6001
Nonlinear feedback	0.1630	0.7710	0.5945
Cascaded systems controller	0.5154	0.6758	0.4568
Stable tracking controller	3.3460	2.9895	8.9369
Full state linearized via dynamic feedback	0.0192	0.0410	0.00017

5.3.2 Test 2 (Straight Line Trajectory)

In the second test, the desired trajectory was defined as follows.

$$x_d(t) = 0, y_d(t) = t \quad (5.5)$$

The desired trajectory of Eq. 5.5 begins at the origin (0, 0) and is a straight line parallel to y-axis. This trajectory is shown for $T = 500$ sec in Figure 5.14. The norm of the errors for the actual trajectory by the leader robot using approximate linearized controller, nonlinear controller, cascaded systems controller, stable tracking controller and full state linearized via dynamic feedback controller is shown in Figure 5.15, 5.16, 5.17, 5.18 and 5.19, respectively. The same values of Table 5.1 for the parameters of the feedback controllers were used.

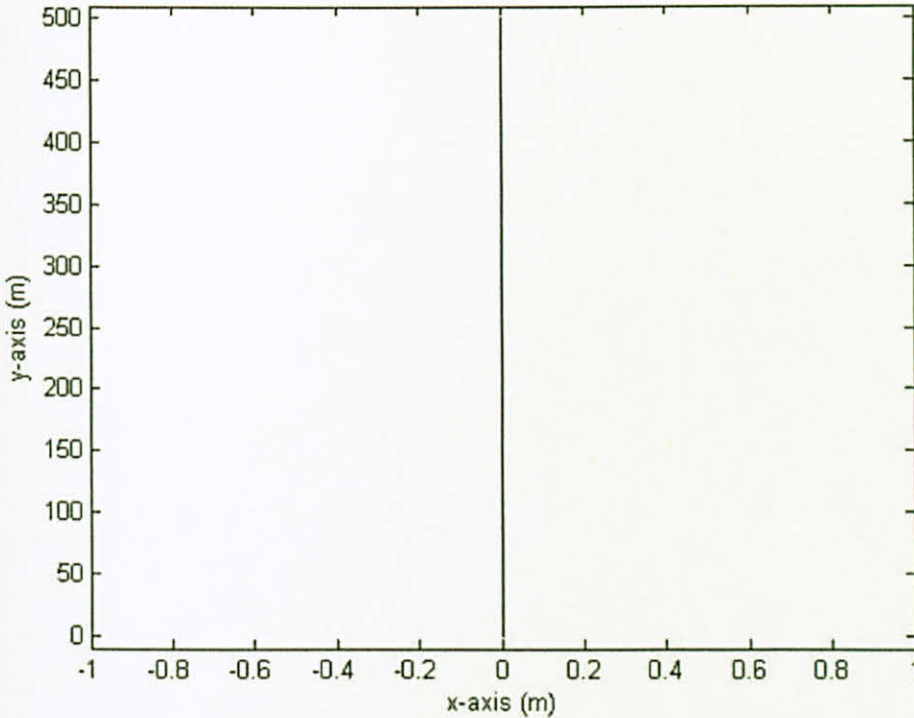


Figure 5.14: Desired trajectory for test 2

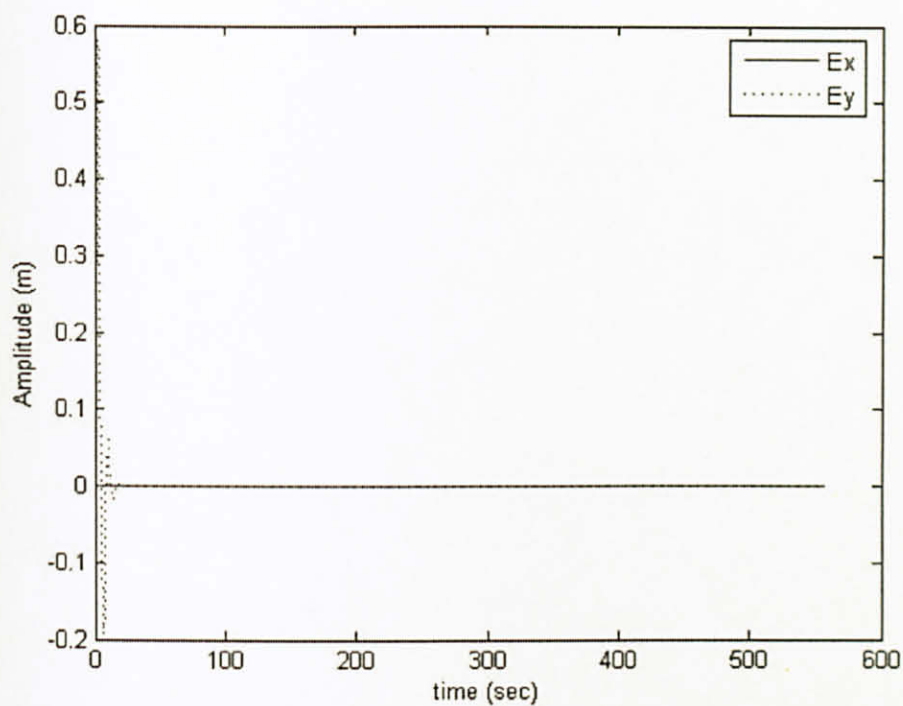


Figure 5.15: Norm of error for the trajectory of test 2 using approximate linearized controller

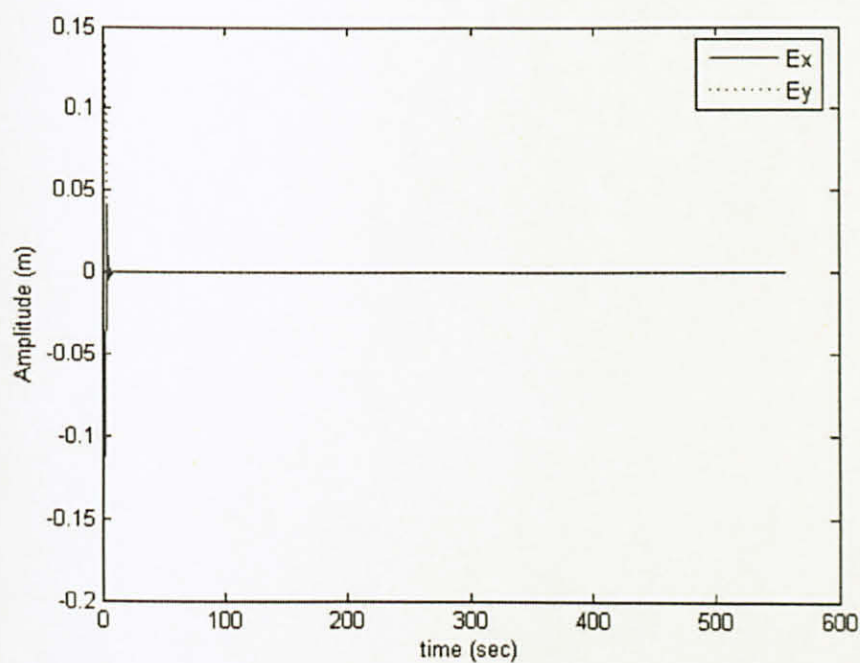


Figure 5.16: Norm of error for the trajectory of test 2 using nonlinear controller

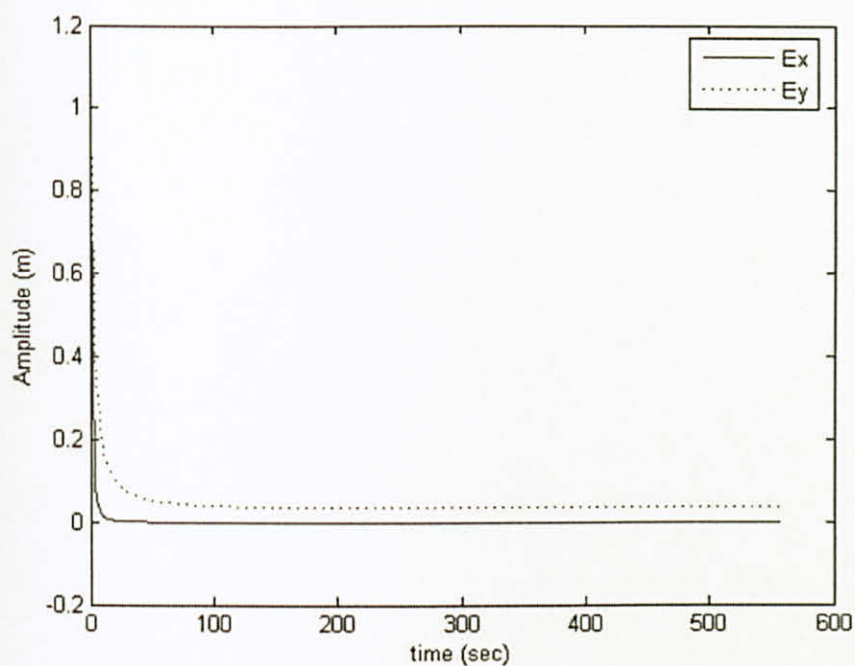


Figure 5.17: Norm of error for the trajectory of test 2 using cascaded systems controller

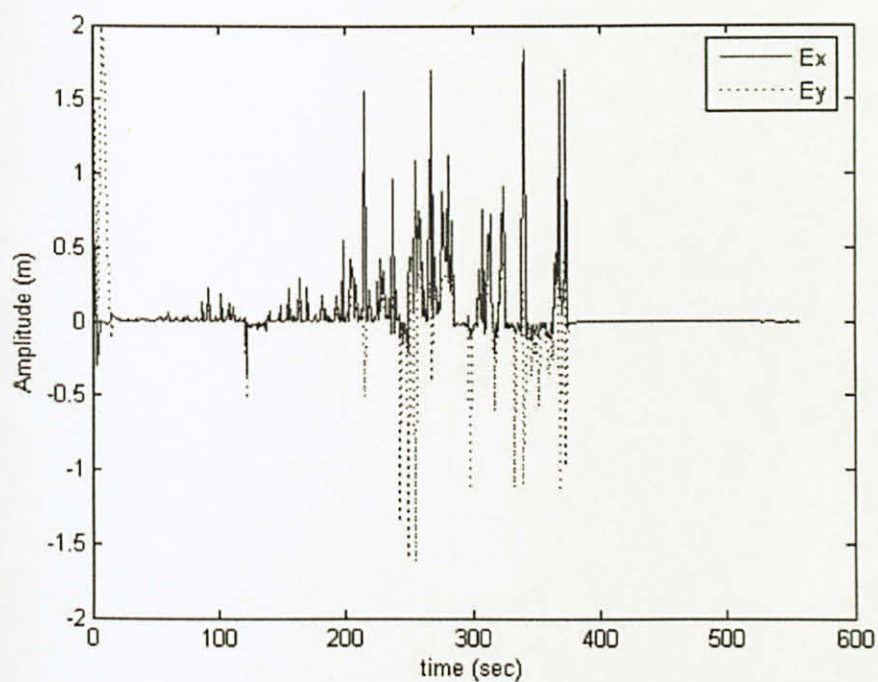


Figure 5.18: Norm of error for the trajectory of test 2 using stable tracking controller

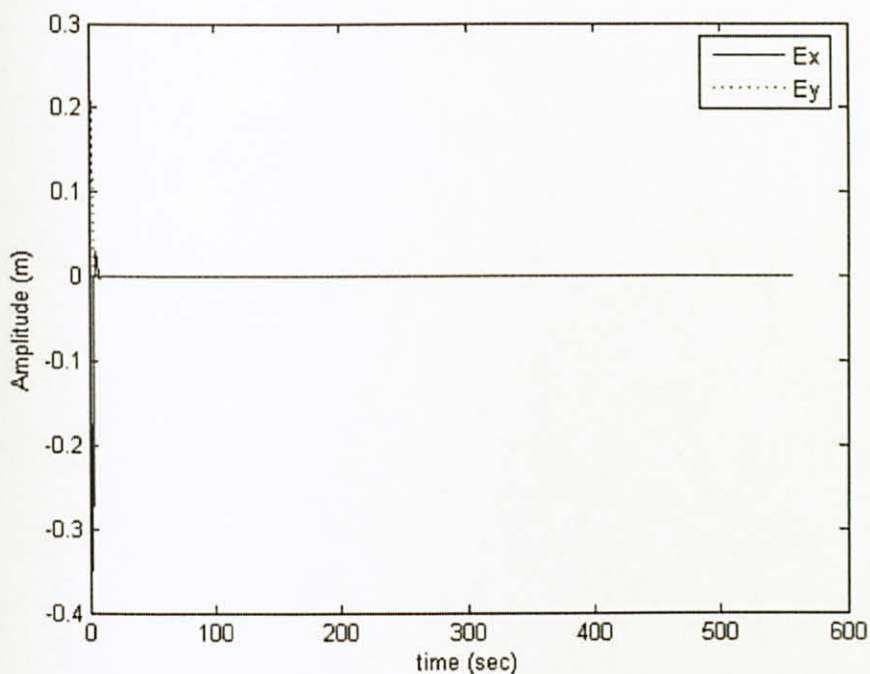


Figure 5.19: Norm of error for the trajectory of test 2 using full state linearized via dynamic feedback controller

Table 5.3 summarizes the error statistics for the given trajectory using different feedback controllers.

Table 5.3: Error statistics using different feedback controllers for test 2

Statistical parameter	Mean (m)	Standard Deviation (m)	Variance (m)
Approximate linearized	0.0222	0.1650	0.0272
Nonlinear feedback	0.0197	0.1558	0.0243
Cascaded systems controller	0.0460	0.0710	0.0050
Stable tracking controller	0.2894	0.5222	0.2726
Full state linearized via dynamic feedback	0.0058	0.0498	0.0025

5.3.3 Test 3 (Sinusoid Trajectory)

In the third test, the desired trajectory was defined as follows.

$$x_d(t) = t, y_d(t) = 10\sin(t) \quad (5.6)$$

The desired trajectory of Eq. 5.6 begins at the origin (0, 0) and is a sinusoidal signal. This trajectory is shown in for $T = 1000$ sec in Figure 5.20. The same values of Table 5.1 for the parameters of the feedback controllers were used. Table 5.4 summarizes the error statistics for the given trajectory using different feedback controllers. The actual trajectory using full state linearized via dynamic feedback linearized controller is shown in Figure 5.21.

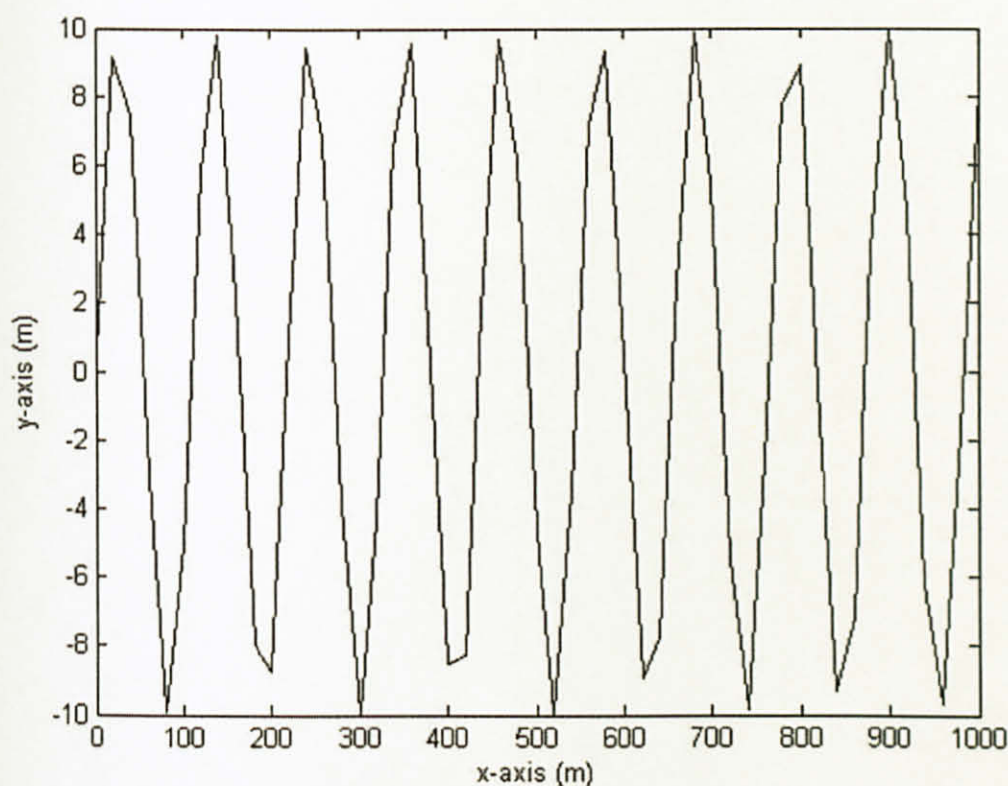


Figure 5.20: Desired trajectory for test 3

Table 5.4: Error statistics using different feedback controllers for test 3

Statistical parameter	Mean (m)	Standard Deviation (m)	Variance (m)
Approximate linearized	0.2808	0.2619	0.0686
Nonlinear feedback	0.3387	0.4052	0.1642
Cascaded systems controller	0.6908	0.6538	0.4274
Stable tracking controller	1.0406	0.8041	0.6465
Full state linearized via dynamic feedback	0.2265	0.6846	0.4687

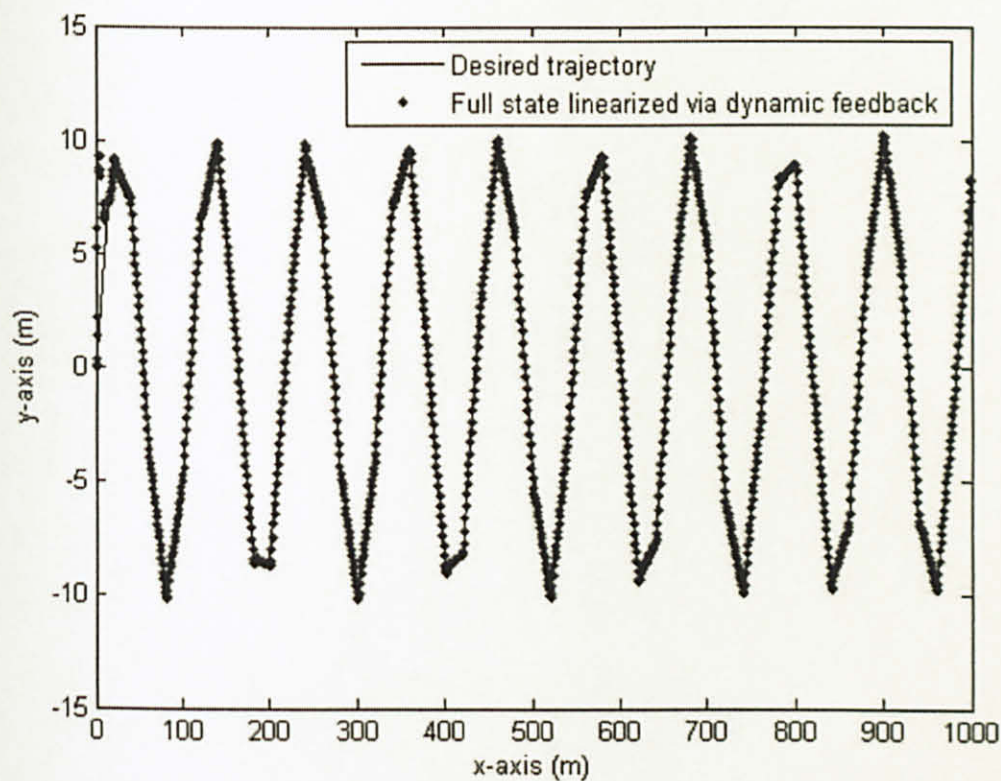


Figure 5.21: Desired and actual trajectory using full state linearized via dynamic feedback controller for test 3

5.3.4 Test 4 (Circular Shaped Trajectory)

In the fourth test, the desired trajectory was defined as follows.

$$x_d(t) = 10 \cos(t/20), y_d(t) = 10 \sin(t/20) \quad (5.7)$$

The desired trajectory of Eq. 5.7 begins at the origin (10, 0) and completes a full cycle when $T = 2\pi(20) \approx 125.67$ sec. The leader robot is assumed to be at the origin (0, 0). The desired trajectory and the actual trajectory for the leader robot using cascaded systems controller is shown in Figure 5.22. The actual trajectory using approximate linearized controller and controller based on Lyapunov function, dynamic feedback controller and nonlinear controller is shown in Figure 5.23 and 5.24, respectively. Table 5.5 summarizes the error statistics for the given trajectory using different feedback controllers.

Table 5.5: Error statistics using different feedback controllers for test 4

Statistical parameter	Mean (m)	Standard Deviation (m)	Variance (m)
Approximate linearized	0.9627	2.4787	6.1438
Nonlinear feedback	1.3260	2.8172	7.9112
Cascaded systems controller	9.7688	0.7949	0.6319
Stable tracking controller	11.3544	1.3286	1.7651
Full state linearized via dynamic feedback	1.0957	2.8231	7.9700

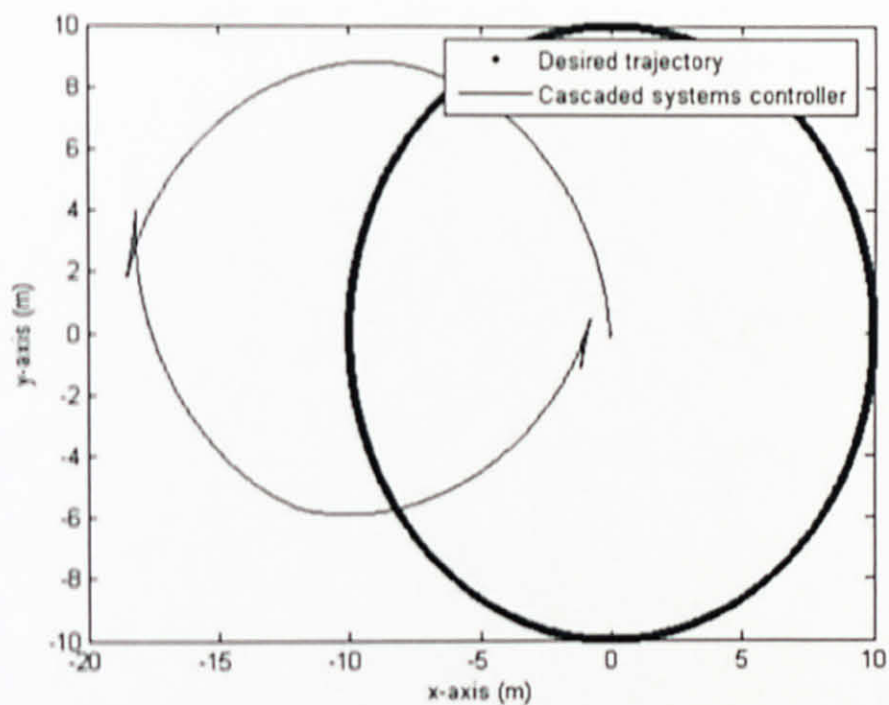


Figure 5.22: Desired and actual trajectory using cascaded systems controller for test 4

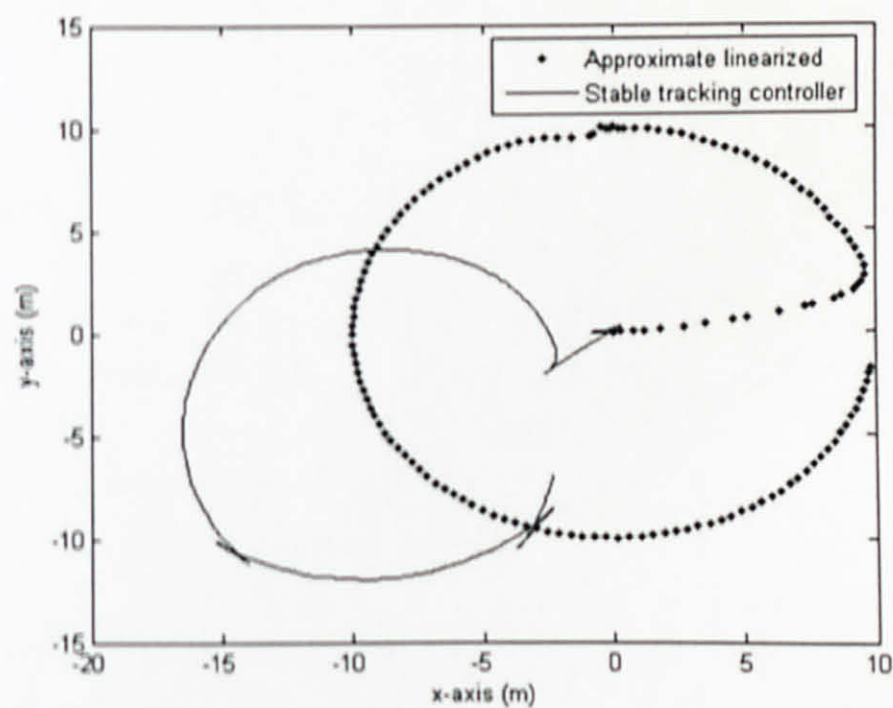


Figure 5.23: Actual trajectory using approximate linearized and stable tracking controller for test 4

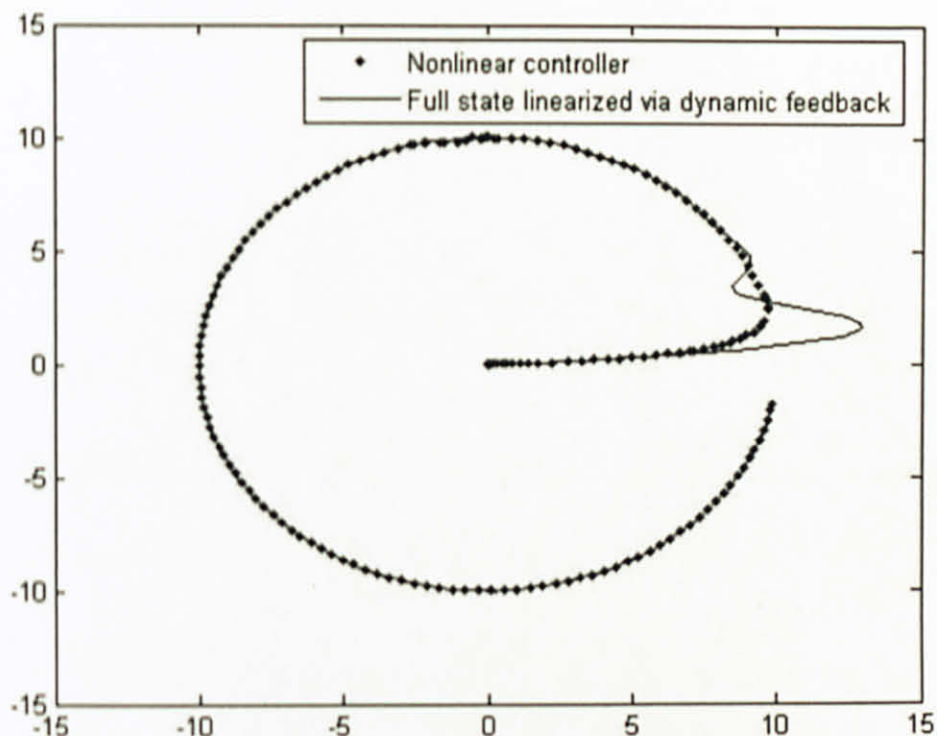


Figure 5.24: Actual trajectory using nonlinear and full state linearized via dynamic feedback controller for test 4

5.4 Discussion of Results for the Leader Robot

Based on these simulation results for the leader robot, it is observed that the full state linearized via dynamic feedback controller minimizes the mean of error more rapidly for the given trajectories. The cascaded systems and stable tracking feedback controllers fail to track the correct trajectory, when the robot and trajectory starting position are not the same. This is observed during test 4 in Figure 5.22 and 5.23.

The reason for failure to track the correct trajectory using cascaded systems controller is that one of the conditions for stability using cascaded systems controller is that ω_d should be persistently exciting. As in trajectory 4, ω_d is not persistently exciting, so the controller can not correctly track the desired trajectory. Using the stable tracking

controller based on linearization of corresponding error model, the system is stable provided $\dot{\omega}_d$ is sufficiently small, which is not the case here. Therefore, the cascaded systems controller and stable tracking controller fail to track the desired trajectory of test 4.

The full state linearized via dynamic feedback controller minimizes the error more rapidly if the trajectory is executed for a long time as observed in test 2 and 3. The effect of changes in the feedback gains and parameters values is shown in Table 5.6.

Table 5.6: Effect of changing gains and parameters in feedback controllers

Feedback controller	Parameter change	Effect of change on mean of error
Approximate linearized	Increase ζ	Decreases
	Increase b	Decreases
Nonlinear feedback	Increase ζ	Decreases
	Increase b	Decreases
Cascaded systems controller	Increase c_1	Decreases
	Increase c_2	Decreases
	Increase c_3	Increases
Stable tracking controller	Increase K_x	Oscillates
	Increase K_y	Increases
	Increase K_θ	Decreases
Full state linearized via dynamic feedback	Increase k_{d1}	Increases
	Increase k_{d2}	Decreases
	Increase k_{p1} till certain value	Increases, then oscillates
	Increase k_{p2}	Decreases

For approximate linearized controller, let ζ be denoted by zeta. The effect of changing the value of *zeta* and the parameter *b* are shown in Figure 5.25 and 5.26, respectively.

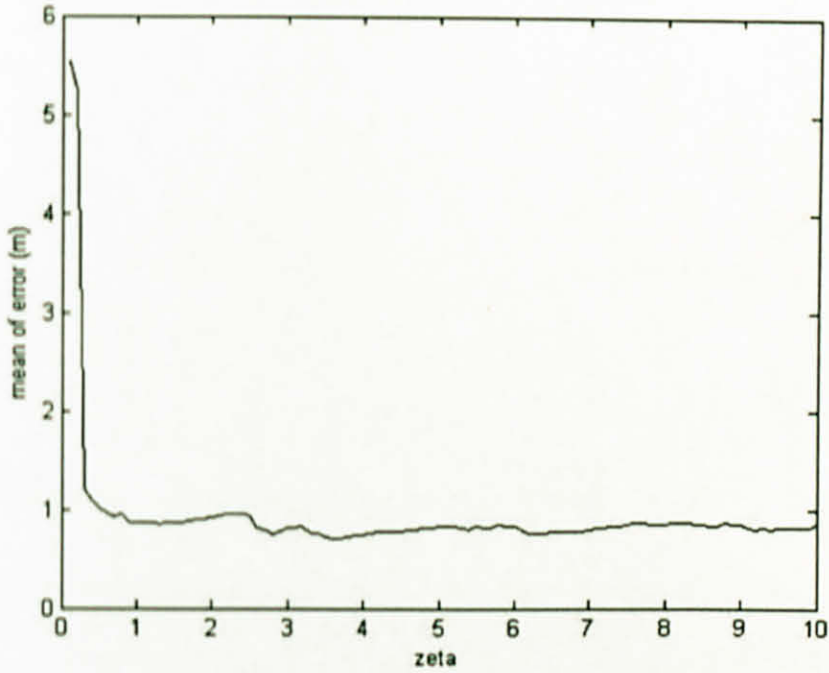


Figure 5.25: Effect of *zeta* on mean of error using approximate linearized controller

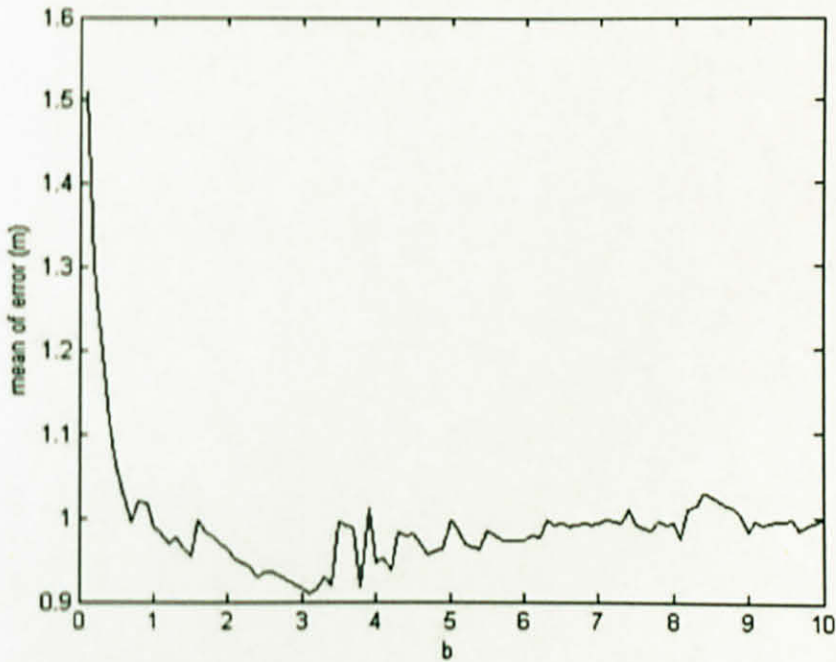


Figure 5.26: Effect of *b* on mean of error using approximate linearized controller

For nonlinear controller, let ζ be denoted by *zeta*. The effect of changing the values of *zeta* and the parameter *b* are shown in Figure 5.27 and 5.28, respectively.

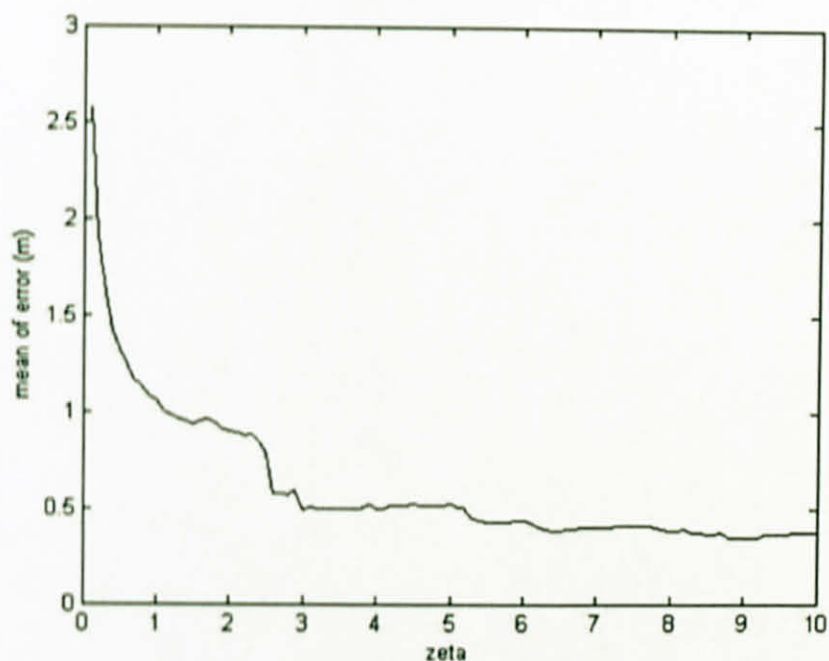


Figure 5.27: Effect of *zeta* on mean of error using nonlinear controller

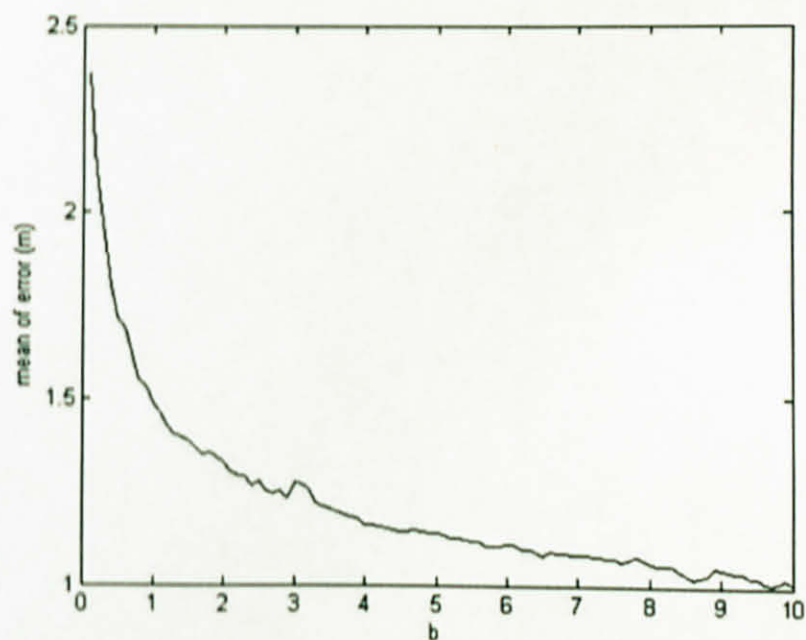


Figure 5.28: Effect of *b* on mean of error using nonlinear controller

For cascaded systems controller, the effects of changing the values of c_1 , c_2 and c_3 on the mean of error are shown in Figure 5.29, 5.30 and 5.31, respectively.

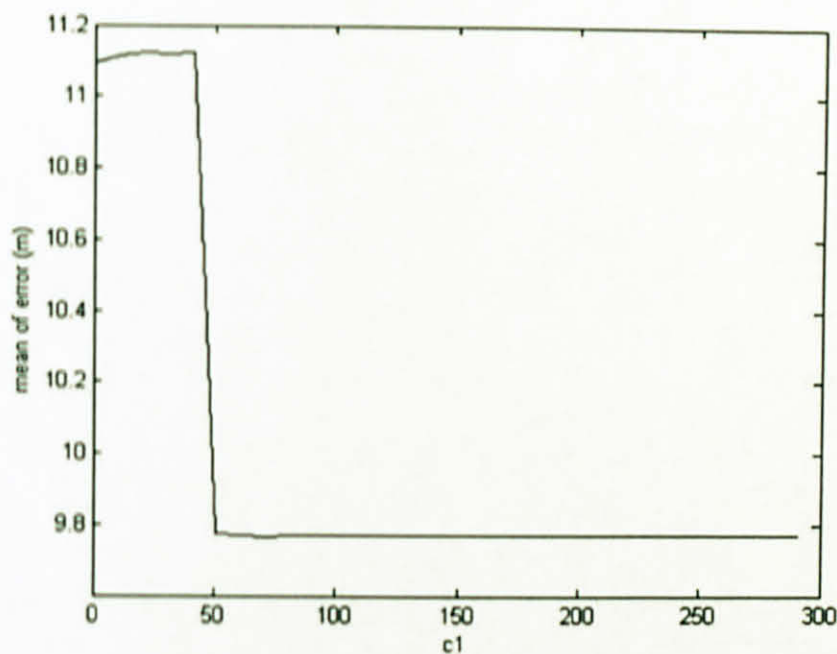


Figure 5.29: Effect of c_1 on mean of error using cascaded systems controller

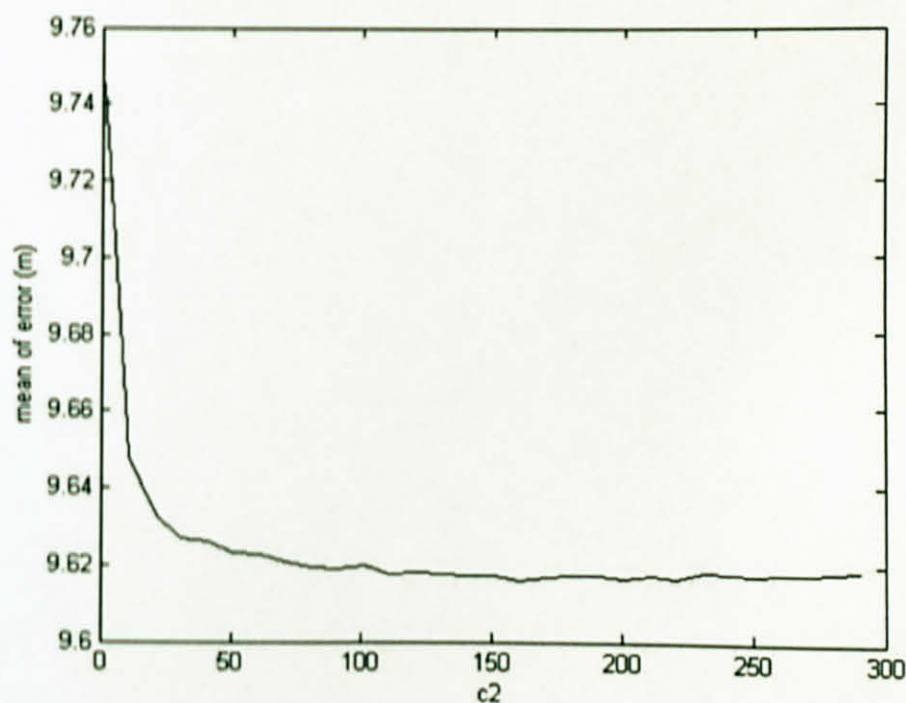


Figure 5.30: Effect of c_2 on mean of error using cascaded systems controller

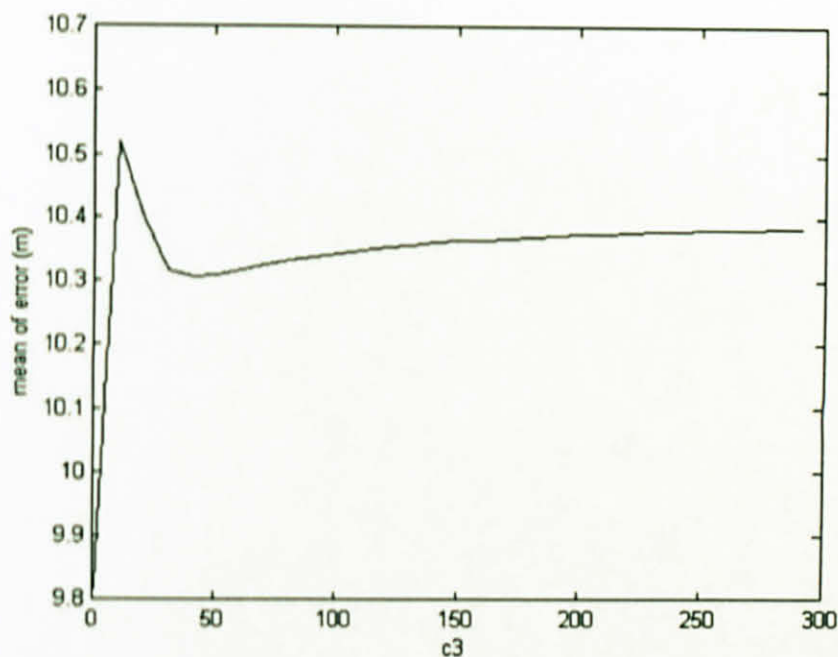


Figure 5.31: Effect of c_3 on mean of error using cascaded systems controller

Using the stable tracking controller, the effects of changing the values of K_x , K_y and K_θ on the mean of error are shown in Figure 5.32, 5.33 and 5.34, respectively.

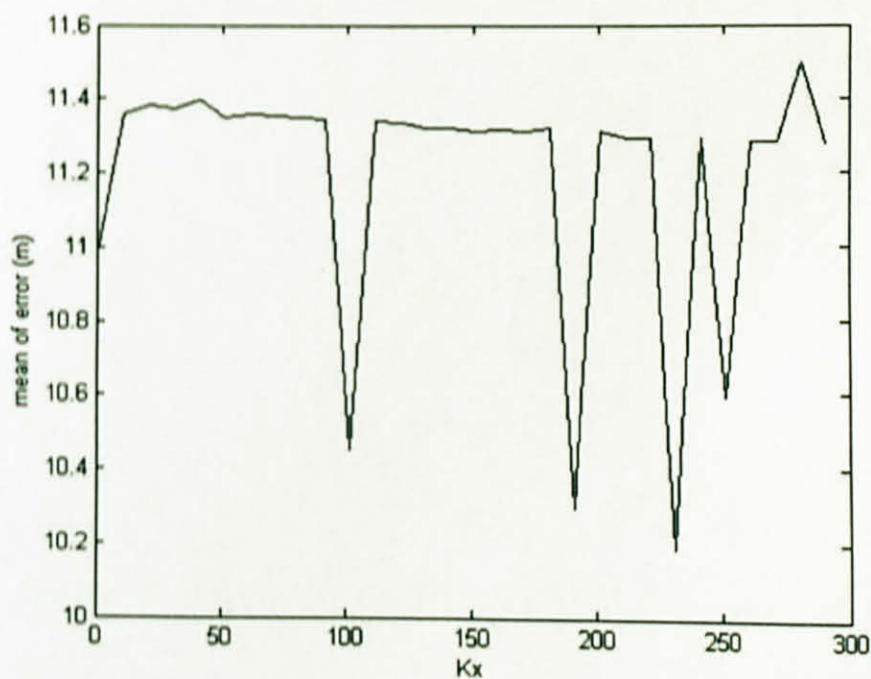


Figure 5.32: Effect of K_x on mean of error using stable tracking controller

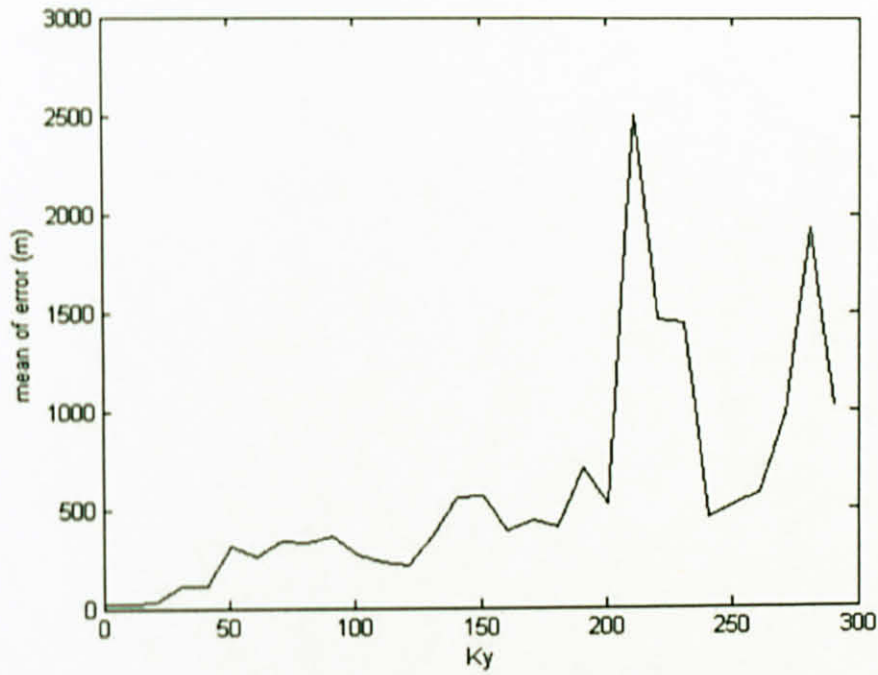


Figure 5.33: Effect of K_y on mean of error using stable tracking controller

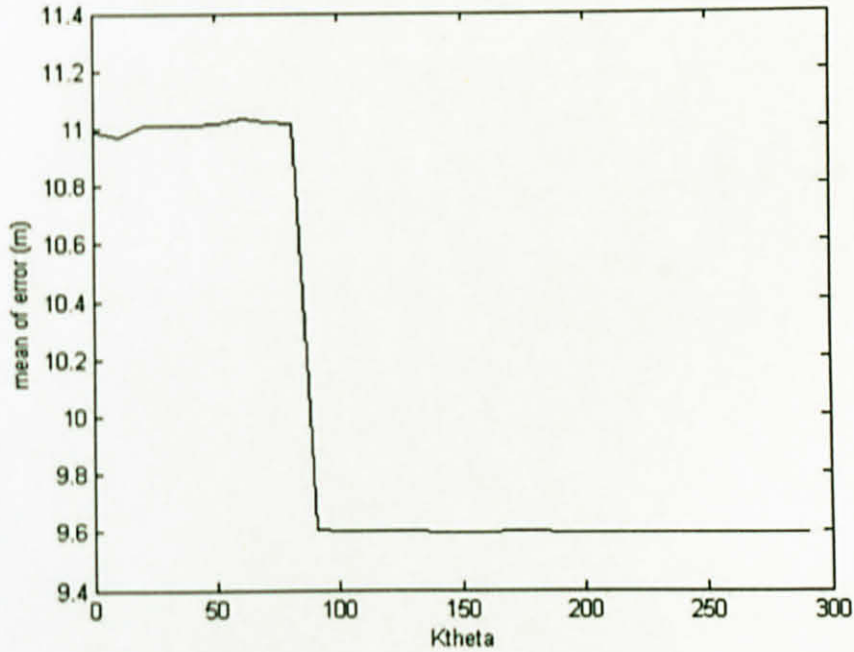


Figure 5.34: Effect of K_{θ} on mean of error using stable tracking controller

Using the full state linearized via dynamic feedback controller, the effects of changing the values of k_{p1} , k_{p2} , k_{d1} and k_{d2} are shown in Figure 5.35, 5.36, 5.37 and 5.38, respectively.

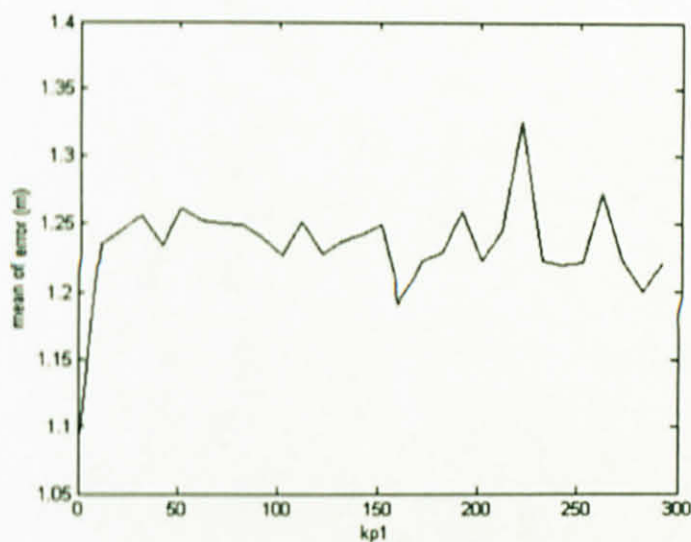


Figure 5.35: Effect of k_{p1} on mean of error using full state linearized via dynamic feedback controller

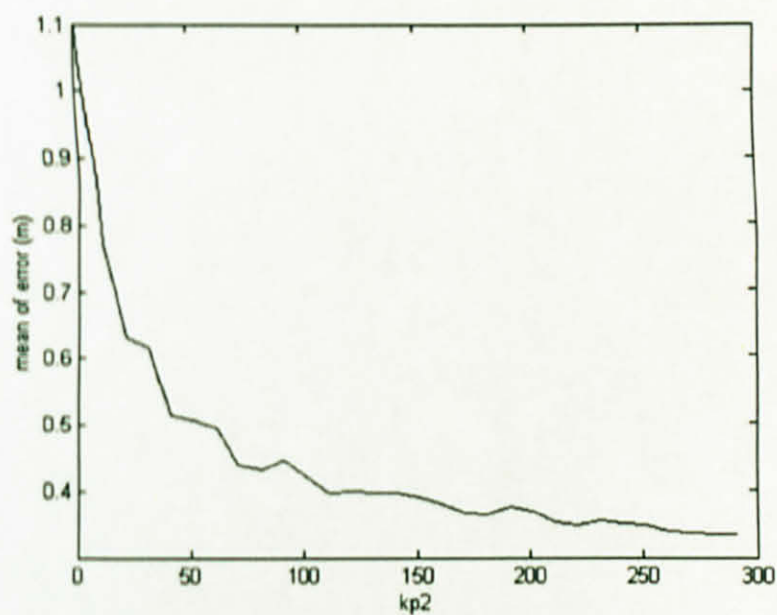


Figure 5.36: Effect of k_{p2} on mean of error using full state linearized via dynamic feedback controller

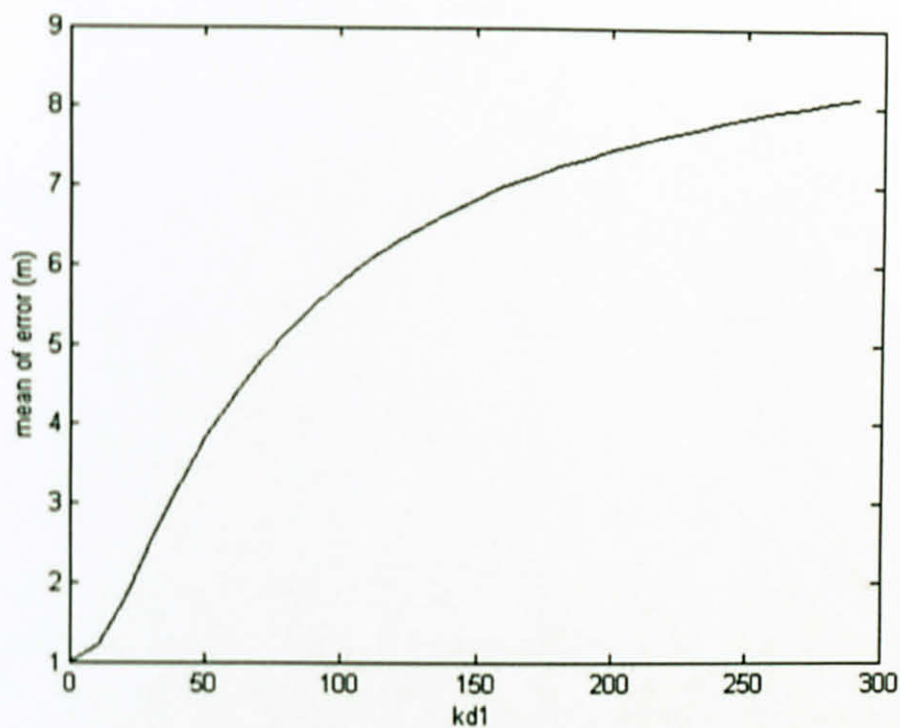


Figure 5.37: Effect of k_{d1} on mean of error using full state linearized via dynamic feedback controller

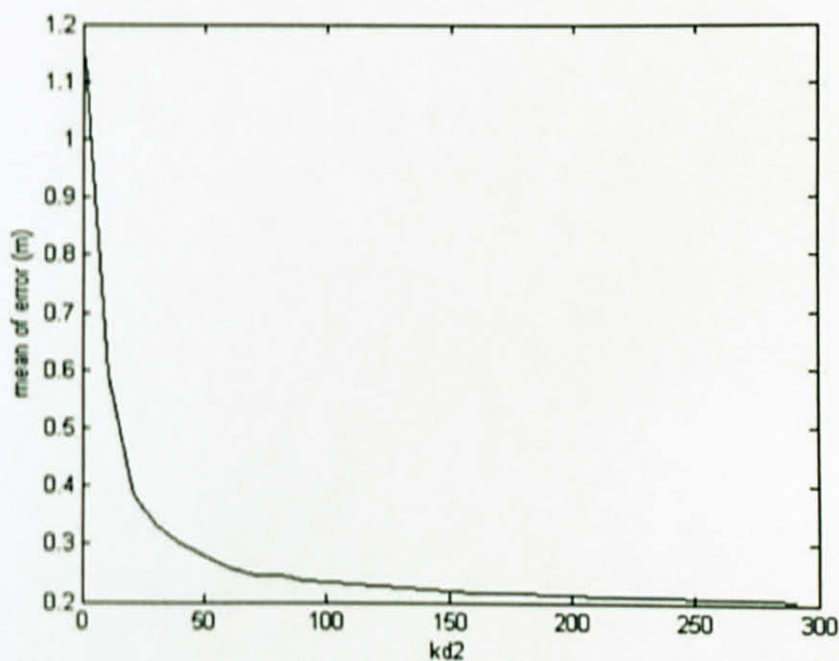


Figure 5.38: Effect of k_{d2} on mean of error using full state linearized via dynamic feedback controller

5.5 Simulation Results for the Follower Robots

5.5.1 Separation Bearing Controller

Using the separation bearing controller, the following parameters were considered.

$$\begin{aligned}
 l_{\eta}^d &= 2 \\
 \varphi_{\eta}^d &= \pi/3 \\
 k_1 &= k_2 = 1 \\
 d &= 1
 \end{aligned} \tag{5.8}$$

The trajectory of test 1 (eight shaped) and test 4 (circular shaped) were used as the desired reference trajectory for the leader robot. The full state dynamic feedback controller was used by the leader robot. The actual trajectory for the leader-follower formation using test 1 and 4 for separation bearing controller is shown in Figure 5.39 and 5.40, respectively. The separation distance and bearing angle for the follower robot is shown in Figure 5.41 and 5.42, respectively.

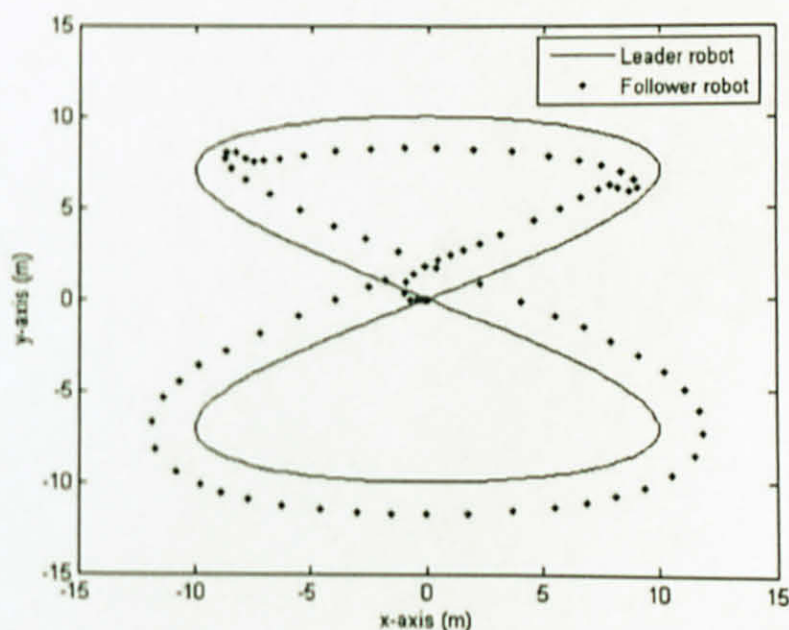


Figure 5.39: Actual trajectory using separation bearing controller for test 1

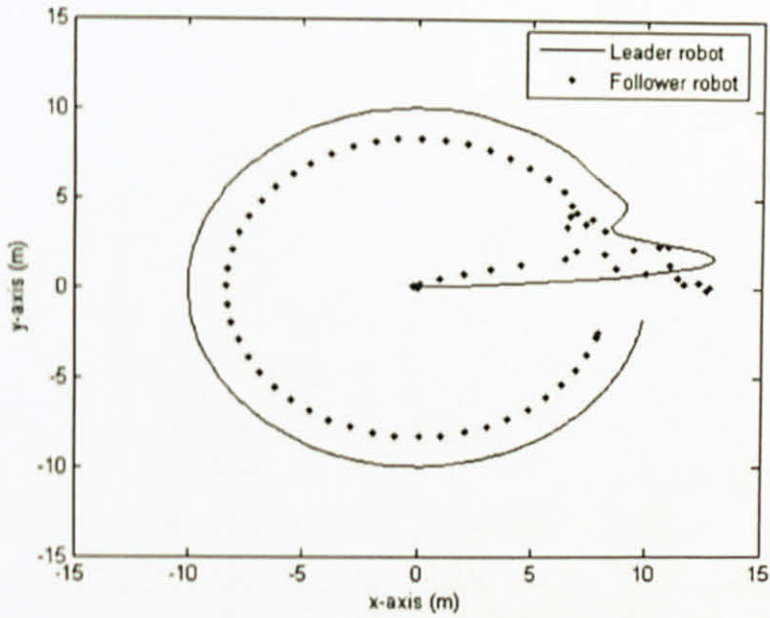


Figure 5.40: Actual trajectory using separation bearing controller for test 4

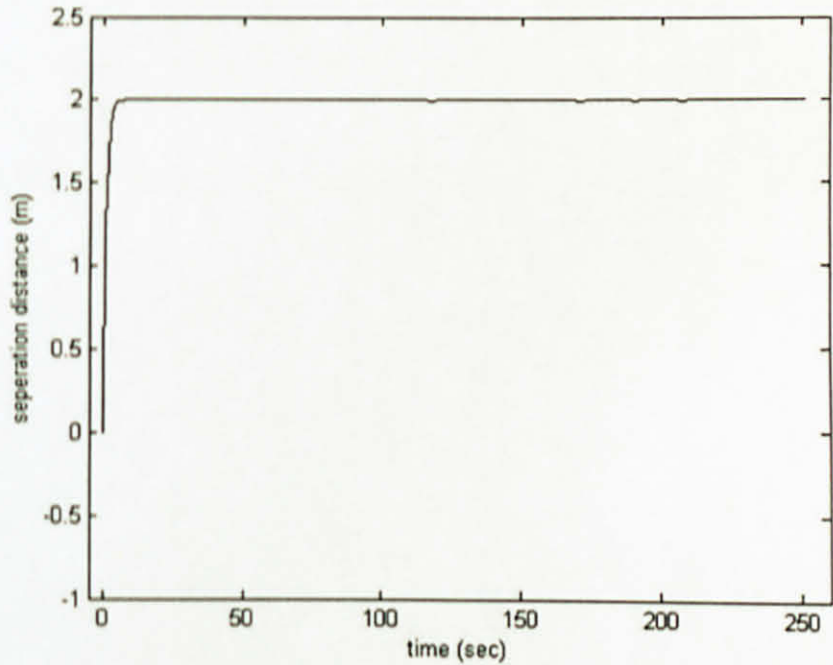


Figure 5.41: Separation distance for the follower robot using separation bearing controller

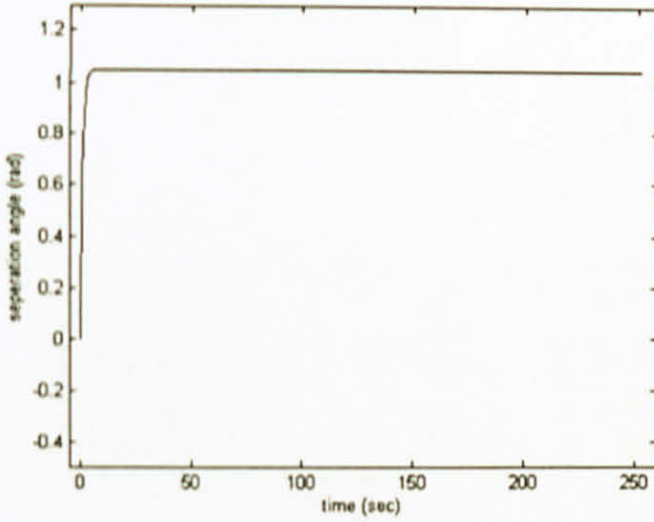


Figure 5.42: Separation angle for the follower robot using separation bearing controller

In another set of simulation, the separation bearing angle was changed as follows.

$$\begin{aligned}\varphi_{ij}^d &= \pi/3 \quad \text{for } t < 100 \\ \varphi_{ij}^d &= \pi + \pi/3 \quad \text{for } t \geq 100\end{aligned}\tag{5.9}$$

The actual trajectory for the leader-follower formation using test 4 is shown in Figure 5.43.

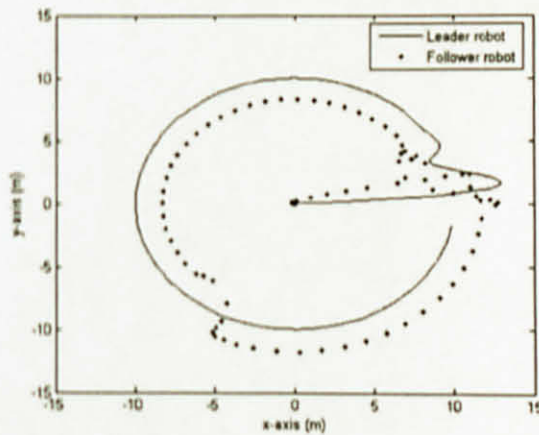


Figure 5.43: Actual trajectory using separation bearing controller for test 4

5.5.2 Separation-separation Controller

Using the separation-separation controller, the following parameters were considered.

$$\begin{aligned}
 l_{1f}^d &= 2 \\
 l_{2f}^d &= 2 \\
 \varphi_{1f}^d &= 3\pi/2 = (240^\circ) \\
 \varphi_{2f}^d &= 2\pi/3 = (120^\circ) \\
 k_1 &= k_2 = 1 \\
 d &= 1
 \end{aligned} \tag{5.10}$$

The actual trajectory for the leader-follower formation using test 1 and 4 for separation-separation controller is shown in Figure 5.44 and 5.45, respectively.

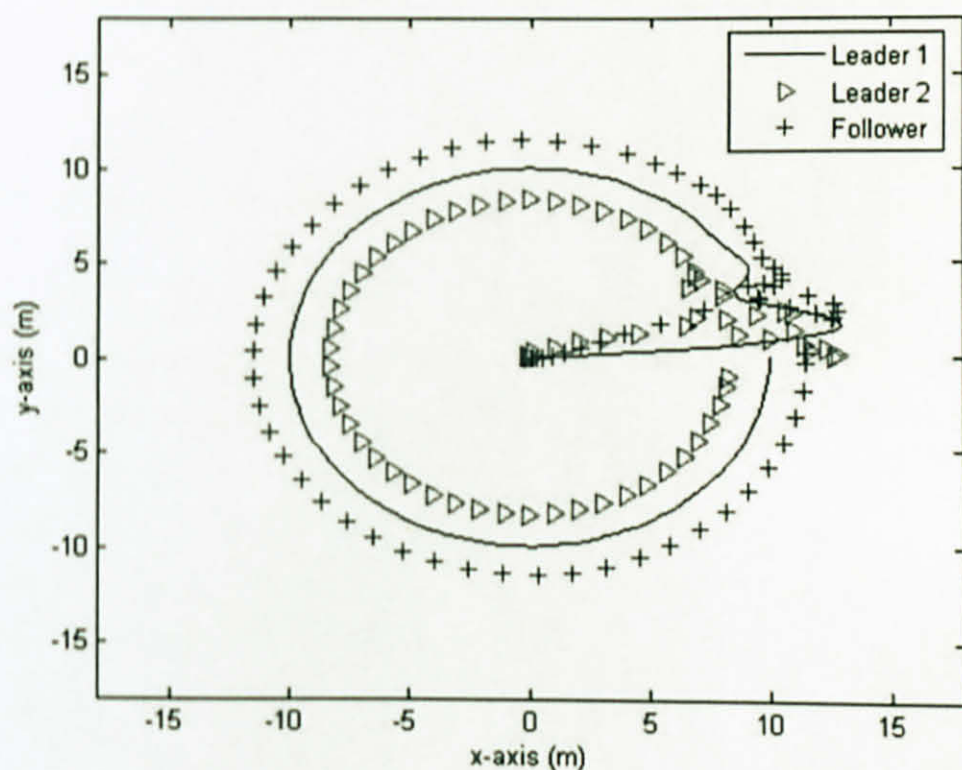


Figure 5.44: Actual trajectory using separation-separation controller for test 1

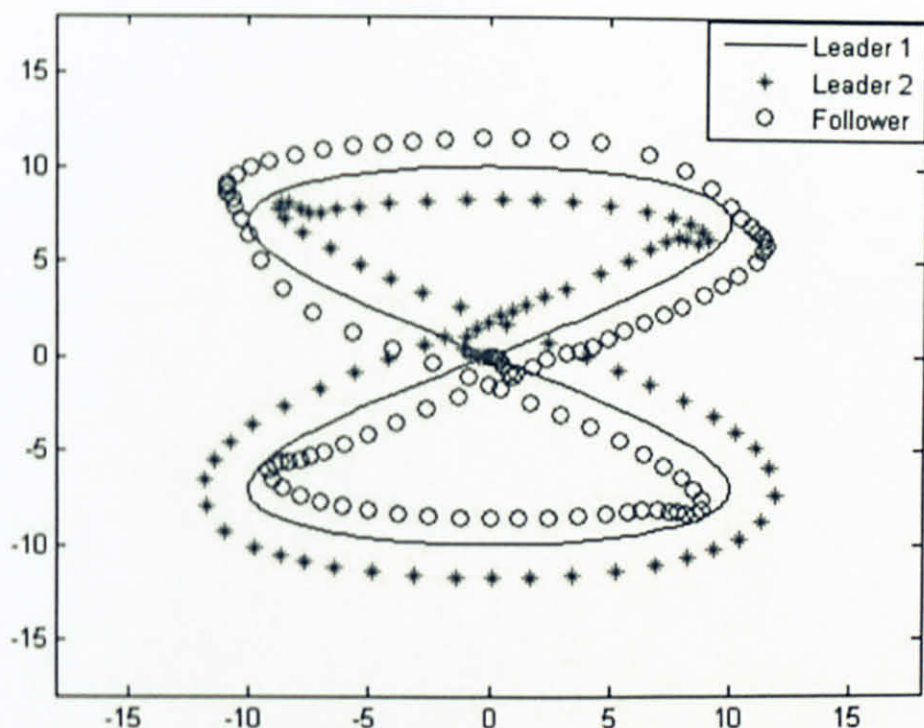


Figure 5.45: Actual trajectory using separation-separation controller for test 4

5.6 Discussion of Results for Follower Robots

Based on the simulation results, it is observed that the input-output feedback linearization for the follower robot minimizes the error between the desired and actual formation. Even, if the parameter values of the separation bearing and separation-separation controllers are changed dynamically at run time, the feedback linearized control strategies successfully minimizes the error between the desired and actual trajectory. Hence, the input-output linearized feedback controller is the preferred controller for separation bearing and separation-separation formation control.

5.7 Posture Stabilization Controller

For posture stabilization, two different goal points were selected as follows. The initial starting position of the leader robot is (-10,-10). The first goal point was defined to reach the origin point (0, 0). The second goal point was to reach the point (-10, 10). The trajectory for the leader robot is not defined. The objective of the leader robot is to move towards the goal point. The results of the posture stabilization controllers are as follows.

5.7.1 Time-varying Feedback Controller

The following parameters were used for the time-varying posture stabilization feedback controller of Eq. 4.51.

$$\begin{aligned}
 y_d(t) &= 0 \\
 \theta_d(t) &= 0 \\
 \omega_d(t) &= 0 \\
 v_d(t) = \dot{x}_d(t) &= -k_5 x_d(t) + g(e, t) \\
 \text{where}
 \end{aligned} \tag{5.11}$$

$$g(e, t) = \frac{\exp(k_6 e_2) - 1}{\exp(k_6 e_2) + 1} \sin t$$

The following values of the gains are used.

$$\begin{aligned}
 k_1 &= 0.5 \\
 k_4 &= 2 \\
 k_3 &= 1 \\
 k_5 &= 1 \\
 k_6 &= 50
 \end{aligned} \tag{5.12}$$

The robot is assumed to be at the point (-10, -10). The results of time-varying posture stabilization controller for the first and second goal points are shown in Figure 5.46 and 5.47, respectively.

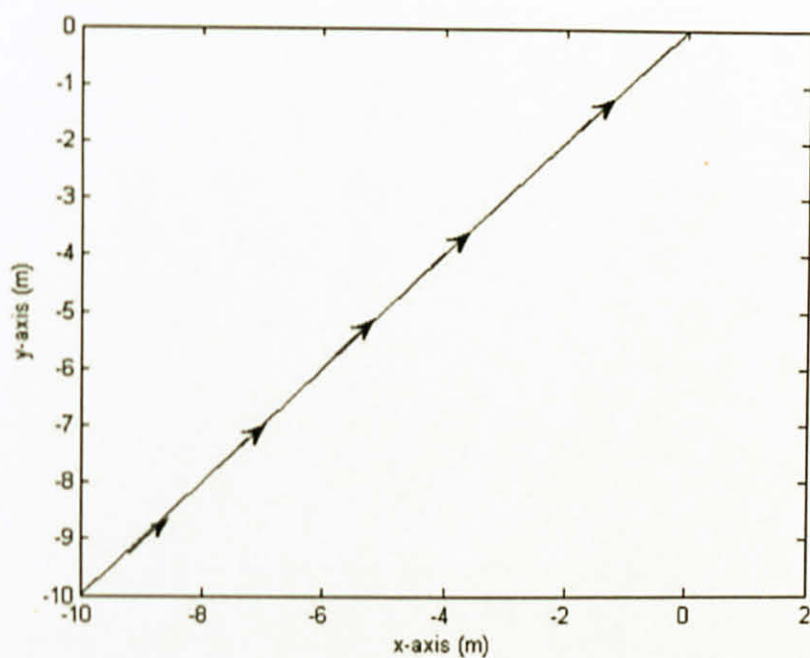


Figure 5.46: Actual point to point motion using time-varying feedback controller for the first goal point.

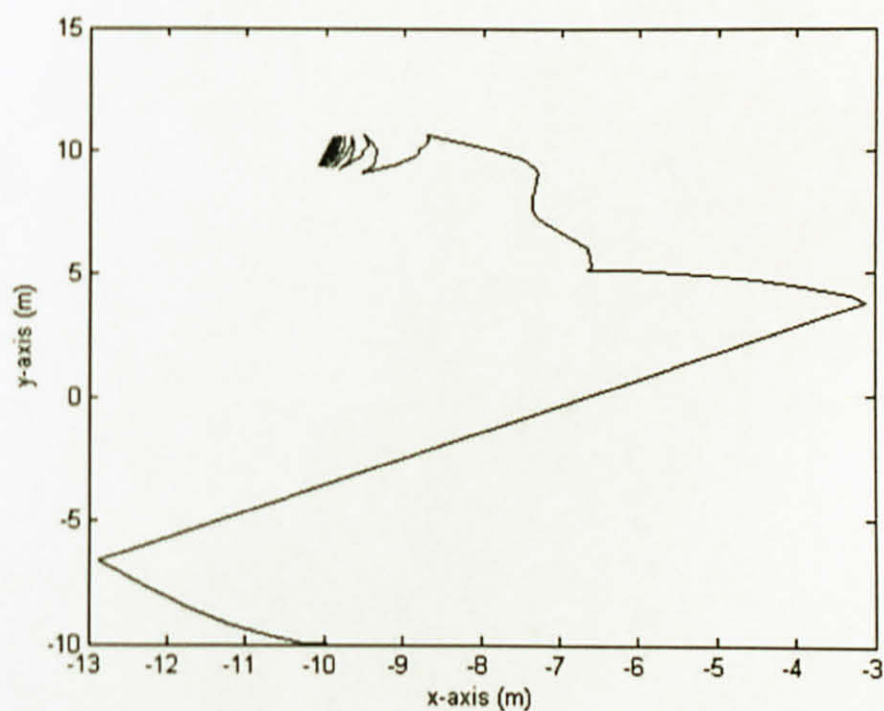


Figure 5.47: Actual point to point motion using time-varying feedback controller for the second goal point.

5.7.2 Polar Coordinates Feedback Controller

The following parameters were used for the polar coordinates posture stabilization controller.

$$\begin{aligned} k_1 &= 1 \\ k_2 &= 3 \\ k_3 &= 2 \end{aligned} \tag{5.13}$$

The robot is assumed to be at the point $(-10, -10)$. The results of point to point motion using this controller for the first and second goal points are shown in Figure 5.48 and 5.49, respectively.

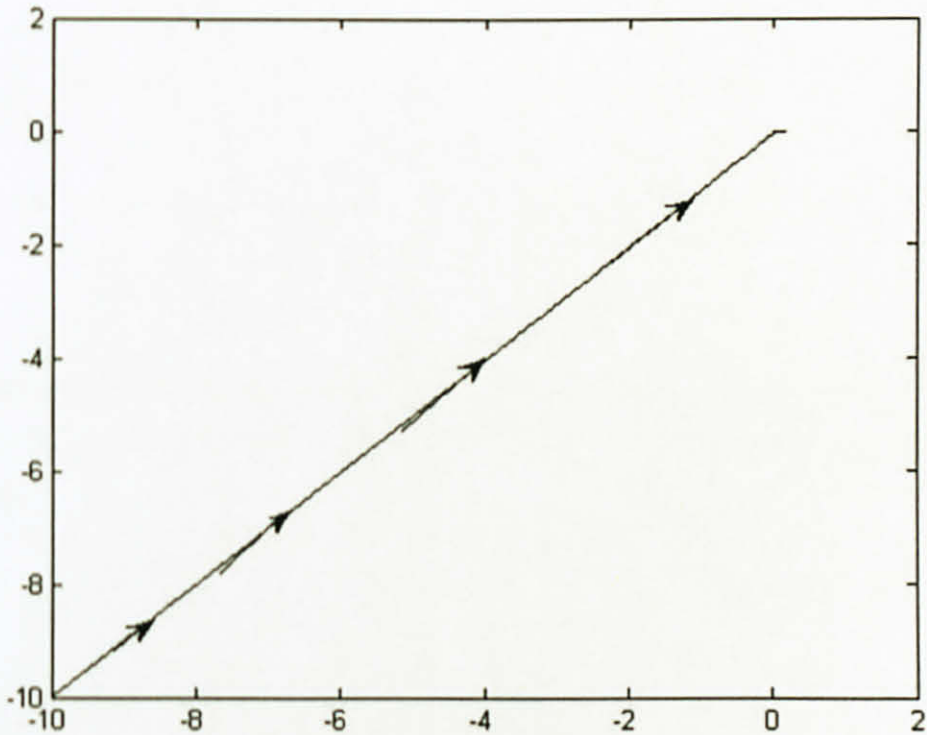


Figure 5.48: Actual point to point motion using polar coordinates feedback controller for the first goal point

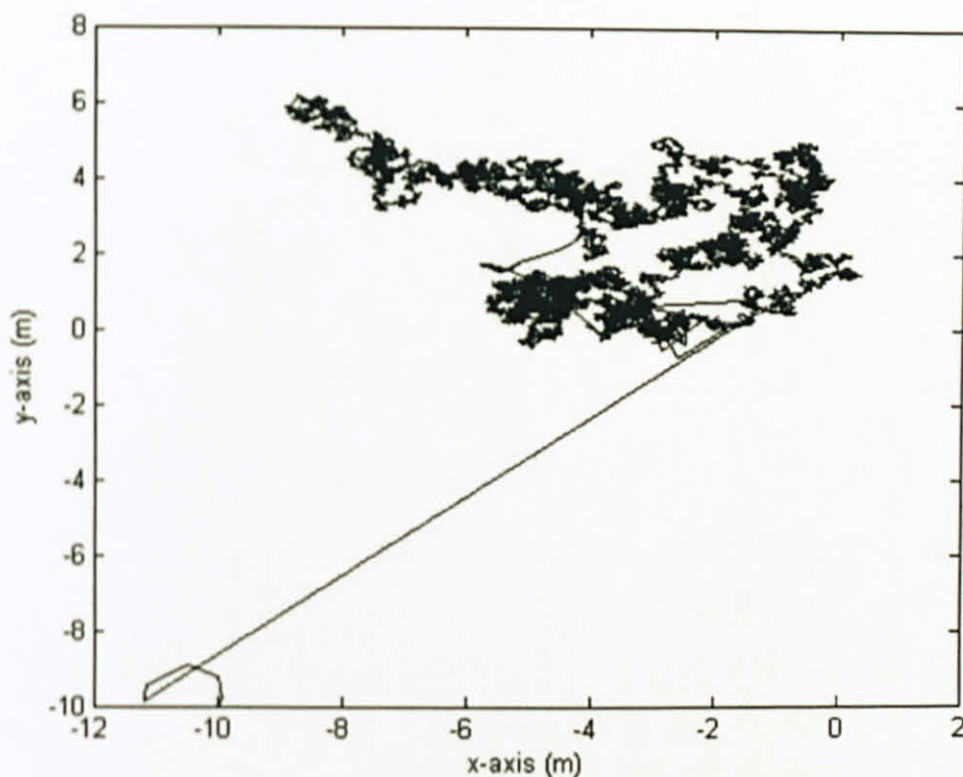


Figure 5.49: Actual point to point motion using polar coordinates feedback controller for the second goal point

5.5.3 Full State Linearized via Dynamic Feedback Controller

The following parameters were used for the full state linearized dynamic feedback controller.

$$\begin{aligned}
 k_{p1} &= 2 \\
 k_{d1} &= 3 \\
 k_{p2} &= 12 \\
 k_{d2} &= 7
 \end{aligned}
 \tag{5.14}$$

The robot is assumed to be at the point $(-10, -10)$ and the goal point is origin. The results of point to point motion using this controller for the first and second goal points are shown in Figure 5.50 and 5.51, respectively.

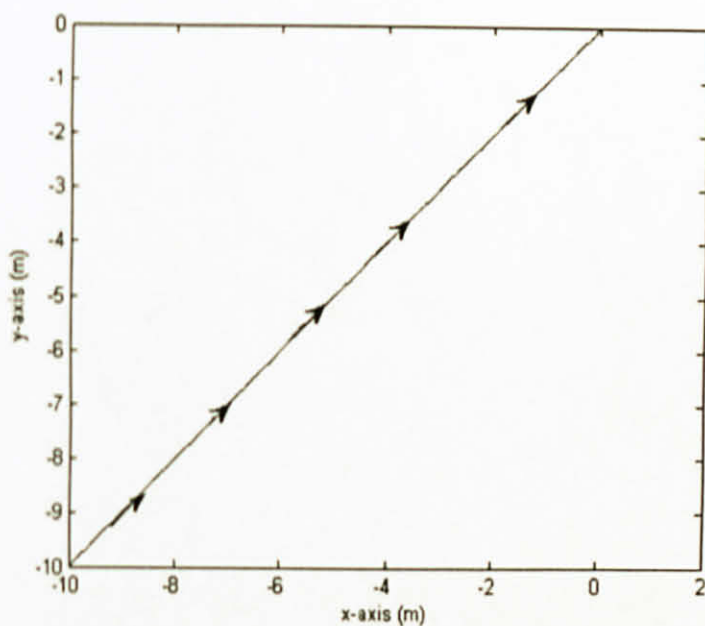


Figure 5.50: Actual point to point motion using full state linearized via dynamic feedback controller for the first goal point

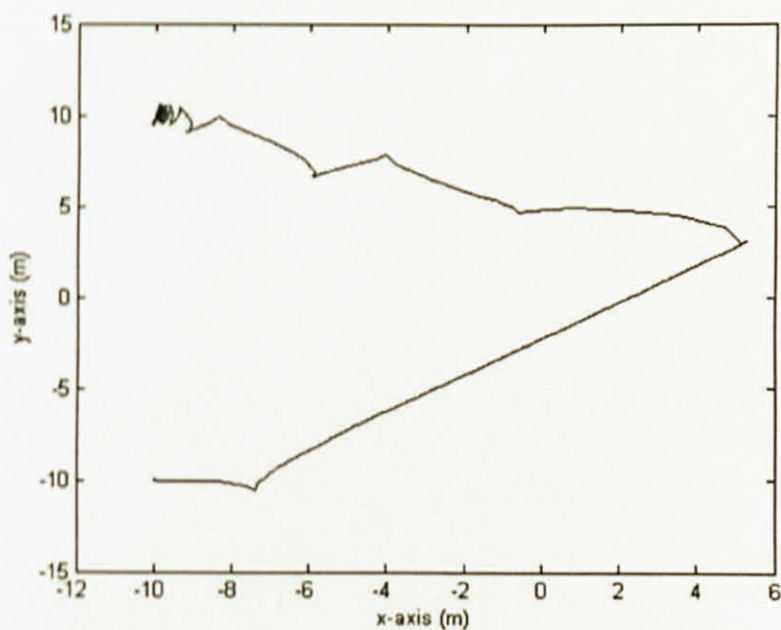


Figure 5.51: Actual point to point motion using full state linearized via dynamic feedback controller for the second goal point

5.8 Discussion of Results for Posture Stabilization

Based on the above results, it is found that the posture stabilization feedback controller based on polar coordinates fails to eliminate the error between the desired and the actual goal point. This can be seen in Figure 5.49 where the desired goal point is $(-10, 10)$. Although the robot is near to the goal point, still it does not converge to the goal point. The robot achieves the correct goal configuration using the nonlinear and dynamic feedback linearized controller.

5.9 Summary

In this chapter, the collaborative robots system was modeled using MATLAB/Simulink. The feedforward controller for the leader robot was derived. The results of different feedback control strategies for the leader robot were compared. The input-output feedback linearized controllers for the follower robot using the separation bearing and separation-separation formation were modeled. Finally the posture stabilization controllers for the leader robot were simulated. In the next chapter, the conclusions and future work are presented based on these simulation results.

CHAPTER 6

CONCLUSION

6.1 Accuracy and Stability of Feedback Controllers

This thesis described the issues related to motion planning of collaborative nonholonomic robots. A kinematic model for the collaborative robots using the leader-follower formation was derived. Control analysis including controllability, stability and observability was performed. The design of feedback controllers for leader-follower formation using feedback linearization techniques was also presented. The follower robots derived their inputs based on the control inputs sent by the leader robot. The leader robot transmitted its control inputs to the follower using the Bluetooth piconet profile.

The posture stabilization controllers using time-varying, polar coordinates and dynamic feedback controller were simulated for the leader robot. The reference trajectory was generated using the feedforward command controller. The collaborative nonholonomic robotic system was modeled using MATLAB/Simulink and the feedback strategies were simulated for a given set of reference trajectories. Based on the simulation results for the various trajectories, the following conclusions are made:

- For the leader robot, the full state linearized controller via dynamic feedback minimizes the mean of error more rapidly than the other feedback strategies.
- The full state linearized dynamic feedback controller for the leader robot achieves posture stabilization.
- The feedback strategies designed using cascaded systems theory and using linearization of corresponding error model fail to track the trajectory if the leader robot's starting point and the trajectory starting point is not the same (circular shaped trajectory).
- The feedback strategy designed using approximated linearization results in a time-varying controller. Hence asymptotic stability is not guaranteed.

- The nonlinear design of feedback control strategy results in global asymptotic stabilization. However, for the given trajectories, the full state linearized via dynamic feedback control strategy minimizes the error more rapidly than the nonlinear strategy. Thus, the full state linearized via dynamic feedback control strategy is preferred over the nonlinear strategy.
- For the follower robot, the input-output feedback linearized controllers minimize the error between the actual and the desired trajectory.
- If the formation structure is changed dynamically at run-time, the input-output linearized feedback controllers minimize the effect of disturbances and errors.

In summary, the feedback linearized techniques for collaborative nonholonomic robots can more rapidly minimize the error for trajectory tracking and achieve posture stabilization. For a given feasible trajectory, the full state feedback linearized strategy for the leader robot and input-output feedback linearized strategies for the follower robots are found to be more efficient in stabilizing the system.

6.2 Thesis Contribution

- In this thesis, a framework for collaborative robots is presented. Unlike most of previous researches, Bluetooth is used as a communication medium for transmitting the leader robot control inputs to the follower robots. In previous research, the follower robots had vision-based capabilities which allowed them to estimate the leader robot position. By allowing robots to communicate using Bluetooth, the exact control inputs of the leader robot are transmitted to the follower robots.
- In this thesis, feedback linearized control strategies are designed for both the leader and the follower robots. In the previous research, feedback linearized strategies have been presented for the follower robots. However, using the leader-follower formation, one of the important aspects is the accurate trajectory tracking of the leader robot. If the leader robot is accurately tracking the desired trajectory, the job of the follower robots is to maintain a relative separation distance and

bearing angle to the leader robot. Therefore, the leader robot plays an important role in the formation control. In this thesis, emphasis is placed on designing feedback linearized strategies for the leader robot and evaluating and comparing different feedback control strategies for the leader robot. Hence, this thesis provides feedback linearized control strategies for all the robots in the formation.

- In this thesis, a complete framework for collaborative robots is presented. A shared library is written using Windows Socket programming and complied using MATLAB compiler. The message format used for communication among the robots conforms to standard Agent Control Language provided FIPA.
- In this thesis, the unicycle model of collaborative robots is considered. The unicycle is the basic model for wheeled mobile robots and cars. The unicycle model of collaborative robots can be extended to complex robotic systems such as underwater robots and flying robots. Hence to design feedback linearized control strategies for complex robotic systems, small modifications are needed in the existing feedback linearized strategies.

6.3 Future Work

- In this thesis, the kinematic model of the collaborative nonholonomic robots has been considered. However, for massive robots and at high speeds, the nonholonomic constraint may not be realistic. It may happen that the robots wheels may slip due to high speed. Hence, the robots dynamics are necessary to be modeled.
- Current implementation of Bluetooth piconet does not support roaming protocol; hence the leadership in the formation is always static. To make the leadership more dynamic, a roaming protocol for Bluetooth can be designed.
- The leader and the follower robots are observable. Based on feedback linearized control strategies, observer based feedback laws can be designed for the leader-follower formation.

PUBLICATIONS

Conferences

1. Salman Ahmed and Mohd N. Karsiti, "A testbed for control schemes using multi agent nonholonomic robots", *IEEE EIT Conference*, May 17-20, 2007, Chicago, USA.
2. Salman Ahmed, Mohd N. Karsiti and H. Agustiawan, "A development framework for collaborative robots using feedback control", *IFAC 3rd International Workshop on Networked Control Systems*, June 20-21, 2007, Nancy University, France.
3. Salman Ahmed, Mohd N. Karsiti and G. M. Hassan, "Feedback linearization for collaborative nonholonomic robots", To appear in *IEEE 2007 International Conference on Control, Automation and Systems*, October 17-18, 2007, Seoul, Korea.
4. Salman Ahmed, Mohd N. Karsiti, J. B. Cruz, "Observer based feedback control strategies for collaborative robots", (Paper submitted for reviewing), *IEEE International Conference on Intelligent and Advanced Systems*, November 25-28, 2007, Kuala Lumpur, Malaysia.

Journal

1. Salman Ahmed, Mohd N. Karsiti and R. N. K. Loh, "Feedback linearization techniques for collaborative nonholonomic robots", (Manuscript under preparation), *International Journal of Control, Automation and Systems*, ICASE, Korea.

REFERENCES

- [1] J. Ferber, *Multi Agent Systems: An Introduction to Distributed Artificial Intelligence*, Addison Wesley Longman, England, 1999.
- [2] L. E. Parker, *Heterogeneous multi-robot cooperation*, PhD Thesis, Department of Electrical and Computer Engineering, Massachusetts Institute of Technology, February 1994.
- [3] A. Bichhi and L. Pallottino, "On optimal cooperative conflict resolution for air traffic management systems", *IEEE Transactions on Intelligent Transportations Systems*, vol. 1, no. 4, December 2000.
- [4] Yi Guo, L. E. Parker and R. Madhavan, "Towards collaborative robots for infrastructure security applications", *International Symposium on Collaborative Technologies and Systems*, pp. 235-240, San Diego, CA, January 2004.
- [5] I. Kolmanovsky and N. H. McClamroch, "Developments in nonholonomic control problems," *IEEE Control Systems Magazine*, vol. 15, issue 6, pp. 20-36, 1995.
- [6] R. W. Brockett, "Asymptotic stability and feedback stabilization," *Differential Geometric Control Theory*, Brockett R. W., Millman R. S., Sussmann H. J. (eds). pp. 181-191, Boston, MA, USA, 1983.
- [7] J. K. Hedrick and A. Girard, "Feedback linearization", *Control of Nonlinear Dynamic Systems: Theory and Applications*, 2005.
- [8] W. Respondek, *Geometry of static and dynamic feedback*, Lecture Notes, Summer Schools on Mathematical Control Theory, Trieste, Italy, September 2001.
- [9] P. Tabuada, G. J. Pappas and P. Lima, "Motion feasibility of multi-agent formations," *IEEE Transactions on Robotics*, vol. 21, pp. 387-392, June 2005.

- [10] T. Balch and R. C. Arkin, "Behavior-based formation control for multi-robots teams", *IEEE Transactions on Robotics and Automation*, vol. 14, no. 6, pp. 926-939, December 1998.
- [11] K. H. Tan and M. A. Lewis, "Virtual structures for high-precision cooperative mobile robot control", *Autonomous Robots*, vol. 4, pp. 387-403, October 1997.
- [12] D. Langer, J. K. Rosenblatt and M. Hebert, "A behavior-based system for off-road navigation", *IEEE Transactions on Robotics and Automation*, vol. 10, no. 6, December 1994.
- [13] C. Michael Clark, *Dynamic robots network: A coordination platform for multi-robot systems*, PhD Thesis, Department of Aeronautics and Astronautics, Stanford University, June 2004.
- [14] A. Isidori, *Nonlinear Control Systems*, 3rd edition, Springer-Verlag, London, 1995.
- [15] C. M. Soria, R. Carelli, R. Kelly and J. M. I. Zannatha, "Coordinated control of mobile robots based on artificial vision", *International Journal of Computers, Communication & Control*, vol. I, no.2, pp. 85-94, 2006.
- [16] E. Lefeber, J. Jakubiak, K. Tchon and H. Nijmeijer, "Observer based kinematic tracking controllers for a unicycle-type mobile robot", *IEEE Conference on Robotics & Automation*, Seoul, Korea, May 2001.
- [17] Y. Kanayama, Y. Kimura, F. Miyazaki and T. Noguchi, "A stable tracking control method for a non-holonomic mobile robot", *IEEE International Workshop on Intelligent Robots and Systems*, Japan, pp. 1236-1241, November 1991.
- [18] C. C. de Wit, H. Khennouf, C. Samson and O. J. Sordalen, "Nonlinear control design for mobile robots", *Recent Trends in Mobile Robots*, World Scientific Publisher, vol. 11, pp. 121-156.

- [19] C. Samson, "Time-varying feedback stabilization of car-like wheeled mobile robots", *International Journal of Robotics Research*, vol. 10, issue 1, pp. 55-64, February 1993.
- [20] G. Oriolo, A. De Luca and M. Vendittelli, "WMR control via dynamic feedback linearization: design, implementation and experimental validation", *IEEE Transactions on Control Systems Technology*, vol. 10, no.6, November 2002.
- [21] J. P. Desai, J. Ostrowski and V. Kumar, "Controlling formations of multiple mobile robots", *IEEE International Conference on Robotics & Automation*, Belgium, May 1998.
- [22] J. P. Desai, *Motion planning and control of cooperative robotic systems*, PhD Thesis, Mechanical Engineering and Applied Mechanics, University of Pennsylvania, 1998.
- [23] H. Asada and J.-J. E. Slotine, *Robot Analysis and Control*, New York, John Wiley & Sons, 1986.
- [24] M. Aicardi, G. Casalino, and A. Balestrino and A. Bicchi, "Close loop steering of unicycle like vehicles via Lyapunov techniques", *IEEE Robotics and Automation Magazine*, vol. 2, issue 1, pp. 27- 35, March 1995.
- [25] A. De Luca, G. Oriolo and M. Vendittelli, "Stabilization of the unicycle via dynamic feedback linearization", *IFAC Symposium of Robot Control*, pp. 397-402, Vienna 2000.
- [26] R. Morrow, *Bluetooth Operation and Use*, McGraw-Hill, New York, 2002.
- [27] J. Bray and C. F. Sturman, *Bluetooth Connect without Cables*, Second edition, Prentice Hall, New Jersey, 2002.

- [28] Peter Rydesater, MATLAB Central File Exchange, [Accessed 18th June, 2007], <http://www.mathworks.com/matlabcentral/fileexchange/loadAuthor.do?objectType=author&objectId=483407>
- [29] *FIPA Agent Communication Language Specifications*. 2002. [Accessed 18th June 2007], <http://www.fipa.org/repository/aclspecs.html>
- [30] H. K. Khalil, *Nonlinear Systems*, 3rd edition, Prentice Hall, 2002.
- [31] A. De Luca, G. Oriolo and M. Vendittelli, "Control of wheeled mobile robots: An experimental overview", *Lecture Notes*, Dipartimento di Informatica e Sistemistica, Universita degli Studi di Roma, Italy, 2001.
- [32] R. M. Murray and S. S. Sastry, "Nonholonomic motion planning: steering using sinusoids", *IEEE Transactions on Automatic Control*, vol. 38, issue 5, pp. 700-716, May 1993.
- [33] R. N. K. Loh, "Modeling and Control of Drive-by-wire Automatic Throttle Control Systems", *Lecture Notes*, Center for Robotics and Advanced Automation, Oakland University, August 2006.

APPENDIX A: M-file for calculating Lie Bracket

```
% LieBracket.m
% Programmed by: Salman Ahmed.

function [h,F]=LieBracket (g1,g2,q,n)

% function LieBracket
% invoked as LieBracket (g1,g2,q)
% Input: vector-fields g1 and g2, variable q, order n
% Output: F=[g1,g2],...,[g1^n,g2]];

% If input arguments are less than 4, so set n=1
if nargin <4, n=1; end

% Find the length of g1, g2 and q
lg1=length(g1); lg2=length(g2); lgq=length(q);

% If g1 and g2 do not have same dimensions, so display
% derror
if (lg1~=lg2) |(lg1~=lgq),
    error('dimensions of g1 and g2 do not match');
end

h=g2;

% Initially make F an array of zeros equal to the dimensions
of "n"
F=zeros(lg1,n); F=sym(F);

for i=1:n
    h=jacobian(h,q)*g1-jacobian(g1,q)*h;
    h=simplify(h);
    F(:,i)=h;
end
```

APPENDIX B: M-file for calculating Linear Controllability

```
% Linear_Controlability.m
% Programmed by: Salman Ahmed.

function [C,R]=Linear_Controlability(A,B)

% function Linear_Contrability
% invoked as Linear_Contrability (A,B)
% Input: system matrix A, input matrix B
% Output: controlability matrix C, rank of controlability
% matrix R

n=length(A);
C=B;
for i=1:(n-1)
    C=[C      (A^i * B)];
end
R=rank(C);
```


APPENDIX C: M-file for calculating Jacobian Coefficients

```
% Linearized_A.m
% Programmed by: Salman Ahmed.

function [A]=Linearized_A(g1,u1,g2,u2,pd)

% function Linearized_A
% invoked as Linear_A (g1,u1,g2,u2,pd)
% Input: system vector fields g1,g2
%        control inputs u1,u2 and point pd
% Output: linearized system matrix A
A= jacobian(g1,pd)*u1 + jacobian(g2,pd)*u2;
```

APPENDIX D: Coordinate Transformation

Considering a linear system

$$\dot{x} = Ax + Bu \quad (1)$$

We want to change the coordinate from x to z as

$$z = Tx \quad (2)$$

whereas T is the transformation matrix. From Eq. 2, we can have

$$x = T^{-1}z \quad (3)$$

If the system matrix A is time-varying, then the derivative of Eq. 2 with respect to time, t , is obtained as:

$$\dot{z} = \frac{d}{dt}Tx = \frac{\partial T}{\partial t}x + T\frac{\partial x}{\partial t} \rightarrow \dot{z} = \dot{T}x + T\dot{x} \quad (4)$$

Substituting the value of \dot{x} from Eq. 1, we get

$$\dot{z} = \dot{T}x + T(Ax + Bu) \quad (5)$$

Substituting the value of x from Eq. 3, we get

$$\dot{z} = \dot{T}T^{-1}z + T(AT^{-1}z + Bu) = \dot{T}T^{-1}z + TAT^{-1}z + TBu \quad (6)$$

which can be simplified as

$$\dot{z} = (\dot{T}T^{-1} + TAT^{-1})z + (TB)u$$

APPENDIX E (Similarity_Transform.m)

```
% Similarity_Transform.m
% Programmed by: Salman Ahmed.

function [An,Bn]=Similarity_Transform(A,B,T)

% function Similarity_Transform
% invoked as Similarity_Transform(A,B,T,p)
% Input: system matrix A input matrix B
%        Transform matrix T, variable p
% Output: new system matrix An, new input matrix Bn

% syms v omega t
% A=[0 0 -v*sin(omega*t); 0 0 v*cos(omega*t); 0 0 0];
% B=[cos(omega*t) 0;sin(omega*t) 0; 0 1];
% T=[cos(omega*t) sin(omega*t) 0; -sin(omega*t) cos(omega*t)
0; 0 0 1];

syms t;
t_dot=diff(T,t);
t_inv=inv(T);

An=t_dot*t_inv + T*A*t_inv;
An=simplify(An)

Bn=T*B;
Bn=simplify(Bn)
```